The University of Jordan

Authorization Form

I, Reem Ali Moh'd Qadan AlFayez, authorize the University of Jordan to supply copies of my Thesis/Dissertation to libraries or establishments or individuals on request, according to the University of Jordan regulations.

Signature:

Date: 28th April 2011

الجامعة الأردنيـــة
كلية الدراسات العليا

التاريـــخ: / /

نموذج رقم (١٨)
اقرار والتزام بالمعايير الأخلاقية والأمانة العلمية
وقوانين الجامعة الأردنية وأنظمتها وتعليماتها
لطلبة الماجستير

أنا الطالب: ريم علي محمد [illegible] الرقم الجامعي: (٨٠٩٠٨٧٧)

تخصص: أنظمة المعلومات الكليـــــة: الملك عبدالله الثاني لتكنولوجيا المعلومات

عنوان الرسالة: Usage Pattern Discovery and Link Recommendation in Web-based Educational Hypermedia

اعلن بأنني قد التزمت بقوانين الجامعة الأردنية وأنظمتها وتعليماتها وقراراتها السارية المفعول المتعلقة باعداد رسائل الماجستير عندما قمت شخصيا" باعداد رسالتي وذلك بما ينسجم مع الأمانة العلمية وكافة المعايير الأخلاقية المتعارف عليها في كتابة الرسائل العلمية. كما أنني أعلن بأن رسالتي هذه غير منقولة أو مستلة من رسائل أو كتب أو أبحاث أو أي منشورات علمية تم نشرها أو تخزينها في أي وسيلة اعلامية، وتأسيسا" على ما تقدم فانني أتحمل المسؤولية بأنواعها كافة فيما لو تبين غير ذلك بما فيه حق مجلس العمداء في الجامعة الأردنية بالغاء قرار منحي الدرجة العلمية التي حصلت عليها وسحب شهادة التخرج مني بعد صدورها دون أن يكون لي أي حق في التظلم أو الاعتراض أو الطعن بأي صورة كانت في القرار الصادر عن مجلس العمداء بهذا الصدد.

توقيع الطالب: ....ريم الزاهري........ التاريخ: ٢٨ / ٤ / ٢٠١١

تعتمد كلية الدراسات العليا
هذه النسخة من الرسالــــة
التوقيع........ التاريخ...٢٨/٤/٢٠١١

USAGE PATTERN DISCOVERY AND LINK RECOMMENDATION IN WEB-BASED EDUCATIONAL HYPERMEDIA

By
Reem Ali Moh'd Qadan AlFayez

Supervisor
Dr. Ammar Huneiti

This Thesis was Submitted in Partial Fulfilment of the Requirements for the Master's Degree of Information Systems

Faculty of Graduate Studies
The University of Jordan

April, 2011

ii

COMMITTEE DECISION

This Thesis/Dissertation (Usage Pattern Discovery and Link Recommendation in Web-based Educational Hypermedia) was Successfully Defended and Approved on 20/4/2011.

Examination Committee	**Signature**
Dr. Ammar Huneiti (Supervisor) Assist. Prof. of Adaptive Hypermedia	-------------------------
Dr. Nadim Obeid (Member) Prof. of Artificial Intelligence	-------------------------
Dr. "Moh'd Belal" Al-Zoubi (Member) Assoc. Prof. of Data Mining	-------------------------
Dr. Ali Al-Dahoud (Member) Assoc. Prof. of Distributed Systems (Al-Zaytoonah University)	-------------------------

Dedication

This thesis is dedicated to my beloved parents who are the reason behind what I have achieved so far and what I am striving to reach. Thank you for the encouragement and support you always gave me. Thank you for believing in me. You gave me the strength to always strive for better. I am honoured to have you as my parents.

I also dedicate this thesis for the faculty members of King Abdullah II School for Information Technology. I learned a lot from you academically and socially in the past five years. It was my ambition from the tenth grade to get into this faculty. It was more than what I imagined. So thank you for having faith in me. You are the reason behind my excellence in both Bachelor and Master studies.

Acknowledgment

First and foremost, I would like to thank my family for the love and support they offered me throughout the years of my study. Dad, you have always been the one who gave me moral support and encouragement to reach what I have achieved. Mom, you have always worried on my health and comfort during the years of my study even more than I have. Thank you words will not be enough for both of you.

I also owe my deepest gratitude to my advisor Dr. Ammar Huneiti, whose guidance and support from the initial to the final level enabled me to develop high level of knowledge in the subject of interest. This thesis would not have been complete without your guidance. Thank you for putting up with me during this semester.

I also would like to show my gratitude to all my friends and colleagues who always been there for me, listened to my ideas, provided guidance and encouragement when it was mostly needed.

Table of Content

List of Tables

List of Figures

List of Abbreviations

AWEHS	Adaptive Web based Educational Hypermedia Systems
KASIT	King Abdullah II School for Information Technology
JU	Jordan University
CIS	Computer Information Systems
OOP	Object Oriented Programming
GSP	Generalized Sequential Pattern
WWW	World Wide Web
AH	Adaptive Hypermedia
AHS	Adaptive Hypermedia Systems
CRM	Customer Relationship Management

USAGE PATTERN DISCOVERY AND LINK RECOMMENDATION IN WEB-BASED EDUCATIONAL HYPERMEDIA

By
Reem Ali Moh'd Qadan AlFayez

Supervisor
Dr. Ammar Huneiti

Abstract

The Word Wide Web is becoming one of the most preferred and widespread mediums of learning. Most of the current web-based educational systems are still delivering the same navigational options for all students regardless of their navigational history. Having no dedicated guidance to students during their browsing for a web-based hypermedia will limit their capability of engaging and experiencing the most of the material published in the hypermedia.

The aim of this thesis is to utilize the usage data resulted from past students' interaction with an online educational hypermedia in order to discover interesting patterns from these data. These patterns are used to personalize and organize the hypermedia system according to each student's browsing history.

This research presents a hybrid data mining approach to provide topic related recommendations for students based on past navigational data of other students. This approach combines clustering, frequent pattern mining, and association rules in order to discover frequent patterns in clusters of students' navigational paths and exploit these patterns in the recommendation process.

The approach has been tested on log files produced after a group of students navigated an online Java Tutorial hypermedia system. The experiments revealed that the proposed approach has the ability to recommend topic related links for students navigating the hypermedia. Furthermore, the proposed approach proved that a clustering process introduced before applying the association rules process would generate links that have higher confidence i.e. better recommendations.

1. Introduction

1.1. Overview

Adaptive Web is a dynamic field with an expansive range of diverse outcome and a promise to solve and improve several web personalization needs. Applying web personalization to educational hypermedia is very useful since E-learning environment is spreading widely without paying much attention to the specific students' needs and how it might affect them.

Online hypermedia users leave a track for every hit they generate while navigating the hypermedia. Log files store every hit generated by every user. The analysis of log files data is part of web usage mining. This will enable recommending useful and topic related links depending on past users navigation history through the hypermedia.

In this thesis a hybrid data mining approach is presented. This approach combines K-means clustering, Apriori algorithm for frequent pattern discovery and association rules to build a recommender engine. This engine will first reduce the size of data where frequent patterns are discovered and then the frequent patterns are exploited to recommend links for students.

The proposed recommender engine is implemented on a set of students' navigational paths where students are clustered and frequent patterns are discovered in each cluster in order to be used in the recommender engine. To validate this approach, a testing phase is implemented where a set of students' navigational paths are applied to the suggested recommender engine and the recommendations are compared with the actual students' visits.

1.2. Problem Definition

E-learning is a new trend of educational systems that emerged after the growth in information technology use and the spread of internet based technologies. E-Learning is a revolution started to appear two decades ago, and it kept evolving (Mustapasa, et al.,2011). Nowadays, different types of E-learning exist. The focus of this study will be on online hypermedia courses, which is also called web-based educational systems. This type of E-learning is massively used nowadays due to the growing amount of tutorials and books published online.

The vast amount of information provided in such educational systems resulted in the need to aid students find specific, relevant, and precise information that match their individual goals, interests and knowledge. In brief the issue is not to provide students with access to educational courses online, but to personalize it to their individual state of mind.

Without individual guidance, students will be dealing with the complexity of navigational possibilities embedded in the educational hypermedia systems that will result in student disorientation while using the website for achieving the final learning goal. In addition, going through different pages without having the correct guidance will result in cognitive overload for the students using this system.

Recommender systems have become powerful tools in many domains from E-commerce to digital libraries and knowledge management. Some recommender systems have also been applied to Adaptive Web-Based Educational Systems (AWBES) for recommending lessons (learning objects or concepts) that students should study next or for providing courses recommendation about courses offered that contribute to the student's progress towards achieving their learning goals.

1.3. Motivation

Adaptive Web-based Educational Hypermedia System (AWEHS) is an alternative to the traditional "one size fits all" approach in the development of web based educational courses (Brusilovsky, 2003) Adaptive hypermedia systems in general attempts to deliver personalized content to the user, based on information accumulated about this user and other users who visited the same website.

In web-based educational hypermedia system, the adaptation space can be limited since online hypermedia systems contain only web pages. Each page contains links and information, so the adaptation can be either in content and it's called adaptive presentation or in links and it is called adaptive navigation support as distinguished by (Brusilovsky,1996) .

Adaptive navigation support is also called recommendation system. Such systems became powerful tools in different domains especially in E-Commerce and digital libraries like Amazon.com (Mobasher, 2007). These systems when applied to educational web-based hypermedia will provide guidance to students dealing with too many choices and complex navigation structure.

This study will demonstrate how link recommendation can be helpful to guide students through educational websites and allow them to achieve the task in mind by recommending the next useful page(s).

1.4. Proposed Technique

This study is conducted on data collected from log files resulted from students accessing a web-based java tutorial uploaded on a server and accessed by second year students, who are studying object oriented programming courses at King Abdullah II School for Information Technology (KASIT) at Jordan University (JU).

We have obtained 7853 records that represent 438 students visiting 195 hypermedia pages. These students are clustered according to page/user visit matrix build at the data preparation phase. After students clustering, Apriori frequent pattern mining algorithm is applied to discover frequent patterns in each cluster of students, these two steps are performed offline. The recommendation process happens online, when a new student visits the website, his/her prior pages visited are extracted and used to classify the new student into one of the clusters obtained. The links recommended for this student are extracted from the frequent patterns discovered for the matched cluster.

The collected data is divided into a training data and testing data. The training data is used in the clustering and pattern discovery phase, and then the testing data is used to resemble online users and to test the proposed approach and its validity.

1.5. Organization of the Thesis

The thesis will be organized as follows, chapter two introduces the literature review and the related work to this research. Chapter three describes the proposed technique of link recommendation. The experimental results and study analysis outlined in chapter four. Finally, the conclusion and the future work are presented in chapter five.

2. Literature Review

The research described in this thesis is an intersection of several research areas, including, web personalization, web data mining, and adaptive web-based educational hypermedia. Each research area is discussed in this chapter in order to understand the proposed links recommendation technique in this thesis.

2.1. Web Personalization

Web personalization can be defined as the ability to tailor and adjust the content or presentation of a web-based system, by inferring what a user requires based on previous and current interaction of a user and possibly other users with the system (Mobasher, 2007).

In today's world of information overload, many technologies are developed and used to organize and filter the data according to users' interests that will result in making user visit to a website more engaging (Sakkopoulos, et al., 2010).

Web personalization has been applied in different domains including digital libraries, E-learning, E-commerce, and search engines (Gao, et al., 2010). Personalization objectives differ from one domain to another, for example personalizing E-commerce websites aims to earn customers loyalty and win new customers. While in personalizing E-learning systems, the objective of personalization is to guide students through the website in order to achieve their learning objectives and meet their needs. Personalizing educational websites is the domain being investigated in this thesis.

In order to understand the term web personalization, two terms needs to be distinguished which are, personalization and customization that sometimes confuses the reader. Personalization and customization both refer to the ability to provide a user with tailored content or presentation of a website. The two notions can be discriminated by who controls the adjustment decision in the hypermdia content or presentation (Mobasher, 2007).

In customization the user is in control of specifying his preferences by using manual configuration provided by the website, *MyYahoo* is an example of such website. In personalization, user preferences and interests are inferred and updated by the system with possible use of other users' information and without explicit control by the user, such as *Amazon.com* website (Mobasher, 2007).

To make web personalization possible, information about individual users are essential in order to perform the personalization or adaption of a hypermedia system. The process of collecting and maintaining up to date information about users' preferences, interests, and interaction with the website is called user modelling (Brusilovsky, et al., 2007).

User profile or user model is resulted from user modelling process. User profile is a key element for web personalization. It is the place where individuals' information is stored, such as his personal information (e.g. gender, age), his usage behaviour, his interests, and intuitions (Gao, et al., 2010) depending on the approach of personalization applied in the website. These user profiles are used by the website in attempt to satisfy users' future needs, interests and predict what the appropriate next link to visit while navigating the website. After all, personalization can be viewed as prediction problem.

The data stored in user profiles can be collected in two ways, either explicitly or implicitly (Sakkopoulos, et al., 2010).

- Explicitly collected user information: it requires users' direct participation like asking the user to fill up survey forms, rate items, or even provide personal information while interacting with the website.
- Implicitly collected user information: it requires the system to collect user relevant data. The system collects data by monitoring users' behaviour while they navigate through the website. In this way the system will be able to track all users' historical browsing habits.

After collecting the data needed for personalization, different techniques used to apply the personalization on websites and will be elaborated in the last section of this chapter.

Web personalization is viewed as an application of data mining (Mobasher, 2007). Data mining techniques are applied on data collected from users' interaction with a website. For that reason the term web mining emerged. The goal of web mining is to gain better understanding of their visitors and efficiently organize the website

Since user profile stores information about user interaction with the hypermedia, and since the hypermedia is adapted according to these profiles, that means web usage mining will be the type of web mining that is used in this research.

A discussion of web mining in general and web usage mining in particular is provided in the next section of this chapter.

2.2. Web Data Mining

As mentioned in the previous section, we are interested in web mining since we will be discussing web-based educational systems adaptation. Different techniques of data mining are applied in web-based educational systems like clustering, classification, association rules, and sequential pattern mining. For further applications about other applications refer to (Romero, et al.,2007).

Web mining extracts knowledge from website data collected using log files or collected in a specific structure implemented by the designer (Romero, et al.,2007). Web mining can be categorized into three different categories based on which part of the web to mine: web structure mining, web content mining, and web usage mining.

- Web structure mining:

  In this category of data mining we are interested in the structure of hyperlinks within the website itself. It can be used to generate information such as the similarity between different websites (Kosala, et al.,2000).

- Web content mining:

  This category of web mining extracts or discovers knowledge from web page contents (Liu, 1998). It can be used to classify web pages according to their topic automatically.

- Web usage mining:

  Web usage mining can be defined as the discovery and analysis of patterns in data collected from users' interactions with the website (Mobasher, 2007).

  The overall process of web personalization based on web usage mining generally consist of three phases (Romero, et. al., 2007b):

  1. Data preparation.
  2. Pattern discovery.
  3. Link recommendation.

The first two phases are performed offline, while the last phase is performed online. An explanation of each phase activities is presented below.

1. Data pre-processing.

   In this phase web log files, that contain usage data produced from students' navigating the website, will be transformed into appropriate format to be used in the second phase.

   Data pre-processing can be decompose into two main steps with subtasks contained in each step as follows (Raju, et al., 2008):

   1. Web log files cleansing
      - 1.1. Attributes filtering
      - 1.2. Records filtering
   2. Identification
      - 2.1. User identification
      - 2.2. Page identification
      - 2.3. Session identification

2. Pattern discovery.

   This phase uses one or combination of data mining techniques, such as clustering, association rule mining, and sequential pattern mining in order to find frequent patterns of pages visited by a student.

3. Link recommendation.

   Links are recommended using discovered patterns from the previous phase. It will provide hyperlinks connecting the current page to other related pages based on different students' navigational paths.

There is an increasing interest in applying data mining techniques on educational systems (Romero, et al., 2009). In traditional learning environment, the learning process involves observing and interacting with students to estimate the effectiveness of the learning process (Romero, et al., 2007a).

In e-learning environment there is less face-to-face communication between the student and the teacher, hence data mining techniques were combined with web-based educational systems to advice and manage students' way of navigation through online material (Mustapasa, et al., 2011).

Data mining techniques facilitate and enhance the learning process as whole, for both students and educators. Figure 1 illustrates the affect of applying data mining in educational systems on the learning process as whole.

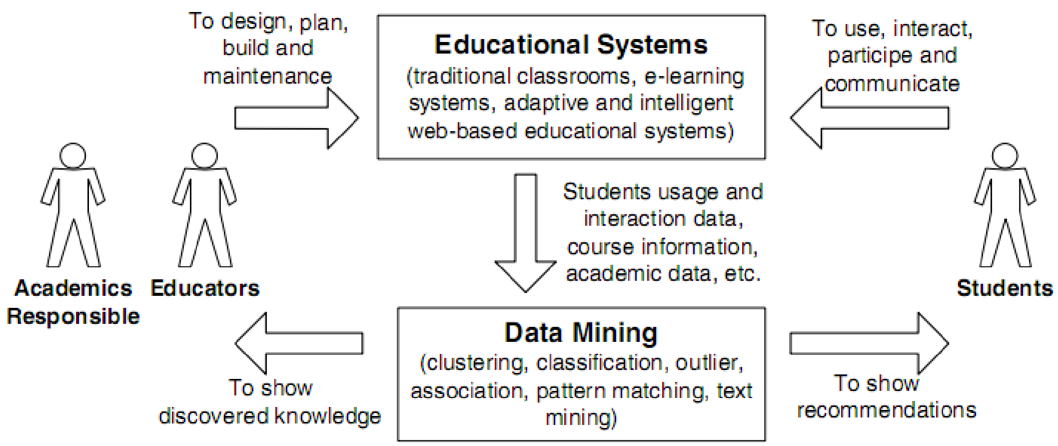


Figure 1: The cycle of applying data mining in educational systems (Romero, et al., 2007a)

There are different web mining techniques applied to educational systems cited in (Romero, et al., 2007a), such as clustering and classification techniques, association rules discovery, and frequent pattern mining.

2.3. Adaptive Web-based Educational Hypermedia Systems (AWEHS)

Adaptive Hypermedia (AH) research area emerged after the maturity of two research areas: Hypermedia and User Modelling. When this area of research came into view, it belonged to adaptive software research area (Brusillovsky, 2001).

The need to have adaptive hypermedia systems became visible after the growth in use of hypermedia. One limitation of static hypermedia systems is that it does not meet the needs of various users' interests, in contrary; it provides the same content and links of web pages to all users.

With the diversity and increasing number of hypermedia users, it is obvious that these systems will have the inability to be "all things to all people" in other words it will not satisfy the audience demands. For example, a newspaper website will recommend the same set of articles to readers with different reading interests.

To solve the above mentioned weakness point in hypermedia systems, adaptive hypermedia approach was developed. It is an alternative to the traditional "one size fits all" approach (Brusilovsky, 2001)

In first years of researching in this new area, the term Adaptive Hypermedia System (AHS) has been defined by one of the pioneer researchers in the field of adaptive hypermedia called Brusilovsky. He defined AHS as "all hypertext and hypermedia systems which reflect some features of the user in the user model and apply this model to adapt various visible aspects of the system to the user" (Brusilovsky, 1996).

By this definition he articulated three criteria that must be in a system, in order to be considered an adaptive hypermedia system. These three criteria are:

1. The system should be hypertext or hypermedia.

2. The system should have a user model.
3. The system should be able to adapt the hypermedia using this model.

2.3.1. Where and why adaptive hypermedia systems can be helpful

AHS are used in different areas, like online information retrieval systems, educational systems, and more.

The area that we are investigating in this thesis is educational systems. Adaptive Educational Hypermedia is a reasonably mature field of study (Brusilovsky, 2003), looking at the researches published and practical systems developed in the past two decades proves the rapid evolving research in this area of adaptive hypermedia systems. The introduction of web-based learning and online tutorials and websites was the driving force behind the maturity of educational hypermedia.

The introduction of e-learning in general was behind having massive amount of educational resources available to students, and that will result in cognitive overload and disorientation in students thinking. Adaptive Educational Systems will help students and guide them through a better learning experience.

2.3.2. Features affects the adaption decision

In AHS, the adaption decisions are based on taking into consideration various aspects like, user characteristics, behaviour or even the environment they work in. In the early researches done on AHS, the adaption to what section, exclusively focused on user characteristics and as AHS has been thoroughly researched, other elements came into view and can be used in the adaption decision.

Kobsa (1999) suggested in his paper that we can distinguish the factors effecting adaption decision into three different categories:

1. User characteristics.
2. Usage data.
3. Environment data.

The third category is relatively new kind of adaption, where the new technologies emerged were the driving force behind its appearance.

Other researchers have different naming for these categories, for example in (Martins, et al., 2008) they divided the adaption to what section into two models that corresponds to the above first two categories.

1. User model
2. Interaction model.

Models are built based on these data, and used to personalize educational websites for students' uses.

2.3.3. What can be adapted

The adaption space in web-based educational hypermedia is limited. What can be adapted? Means what features of the system differs for different users.

Hypermedia systems consist of connected pages, each page contain information and links to related pages in the system, for that reasons, the features that can be adapted are limited to two distinct areas (Brusilovsky, et al. 2003):

- Adaptive presentation (content –level adaption): it adapts the content of the website to the user's goals, knowledge and other information stored in the user model.
- Adaptive navigational support (link level adaption): it supports student navigation in the website by changing the appearance of visible hyperlinks.

This thesis will investigate the second area of adaption, where new links will be recommended to students according to their past navigational path.

There is a lot of adaptive educational hypermedia systems developed since 1996 (Brusilovsky, 2001), such as AHA (De Bra, et al., 1998), InterBook, medtach, Metalinks and more (Romero, et al., 2009). Brusilovsky (2007a) is a survey that describes the different tools developed in adaptive educational hypermedia systems.

2.4. Recommendation in AEHS using Sequential Pattern Mining

Although recommendation and personalized suggestions using data mining techniques, where first introduced and applied in E-Commerce field, to apply the concept of product suggestion based on users' preferences and other consumers purchasing habits. Customer Relation Management (CRM) is crucial issue in today's highly competitive market, because customer satisfaction is very important for companies (Liu, et al.,2009)

Recommendation in E-Learning systems can be useful considering the reasons for using adaptive educational systems in the previous section. Providing links for students to follow and navigate through the educational system is a helpful asset that can be added to every educational system.

Different examples of E-Learning recommendation systems are mentioned in (Aleksandra, et al., 2010), some of the techniques mine the sequence of users' navigation through web pages taking into consideration time of access to each page, while other techniques do not. Different techniques may deal with different kinds of data like temporal databases, relational databases, or transactional databases (Zhao, et al., 2003).

The basic technique for link recommendation system in e-learning is shown in figure 2.

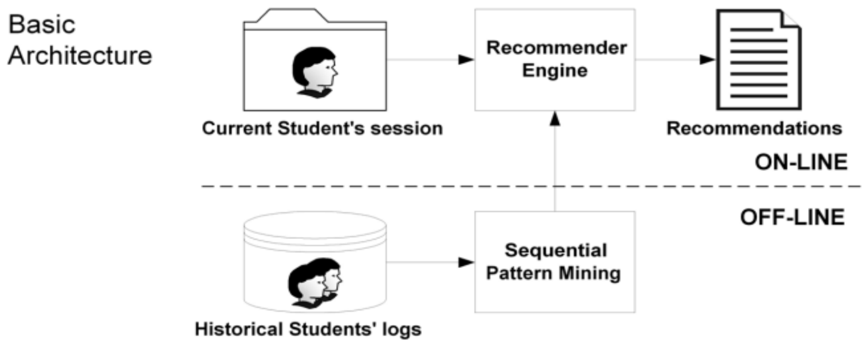


Figure 2: Architecture for recommender system (Romero, et al.,2007b)

This is a simple architecture of recommender system that can be applied on web-based educational systems. It only uses students' usage data extracted from log files and use these data as input to one mining technique, which is in the above figure sequential pattern mining algorithm, to produce frequent patterns that can predict the student most appropriate next page to visit where frequent Patterns are itemsets that appear in a dataset with a frequency no less than a user specified threshold (Han, et al., 2007).

Advanced techniques are proposed to recommend links to students (Romero, et al., 2007b). Such advanced techniques aim to reduce the domain where patterns can be discovered and make the recommendation process faster and more accurate. These advanced techniques integrate several data mining techniques such as clustering, frequent pattern mining, and association rules. It may also use additional information about students other than their interaction with the website.

Examples of such advanced techniques are cited in (Romero, et al., 2009), clustering techniques have been used to find groups of students with similar learning goals and use these groups of students as input for association rules discovery. Also a hybrid of sequential pattern mining with collaborative filtering technique has been used for product recommendation (Liu, 2009).

2.5. Data Clustering

Clustering is finding groups of objects, in our case students, satisfying the objective that the students within a group are similar (related to) one another and different from (or unrelated to) the students in other groups (Tan, et al., 2005).

Clustering has immense number of applications in our daily life, and in information systems field, clustering is often one of the first steps of data mining analysis to discover hidden patterns in huge amount of data.

K-means clustering is one of the oldest and most widely used clustering algorithms (Tan, et al., 2005). K-means clustering technique is simple and easy to implement, and these are the reasons behind being the most widely used clustering algorithm.

The operation of K-means starts by choosing k initial centroids, where k is user defined parameter, which represents the number of clusters desired and directly generate k random points as cluster centers. Then the following steps are applied:

- Assign each point to the closest cluster center, with respect to a distance proximity measure chosen in advance.
- Re-compute the new cluster centers.
- Repeat the two previous steps until centroid values are not changed.

Proximity measures calculate the distance between a point and the cluster centroid. They are used to quantify the closest cluster to each point of data. Some proximity measures, or sometimes called distance functions, are listed below (Liu, 1998):

- Euclidean distance.
- City block distance.
- Cosine distance.
- Hamming distance.
- Jaccard distance.

These distance measures are used according to the type of data being clustered. Also the validity of clusters found determines if a distance function is appropriate, or another distance measure should be used for clustering.

Clusters validation is an importance step in data clustering operation. It measures the goodness or healthiness of clusters obtained from one algorithm, in comparison with clusters obtained by other clustering algorithms. It can also measure the healthiness of clusters obtained from the same algorithm using different parameter values, such as the *k* parameter that represent the number of clusters obtained in K-means algorithm (Tan, et al., 2005).

3. Proposed technique for link recommendation

3.1. Overview

A technique for links recommendation based on web usage mining is proposed. Combinations of different data mining methods are used in this technique and are elaborated in the following sections. Figure 3 presents the proposed technique of link recommendation in details.

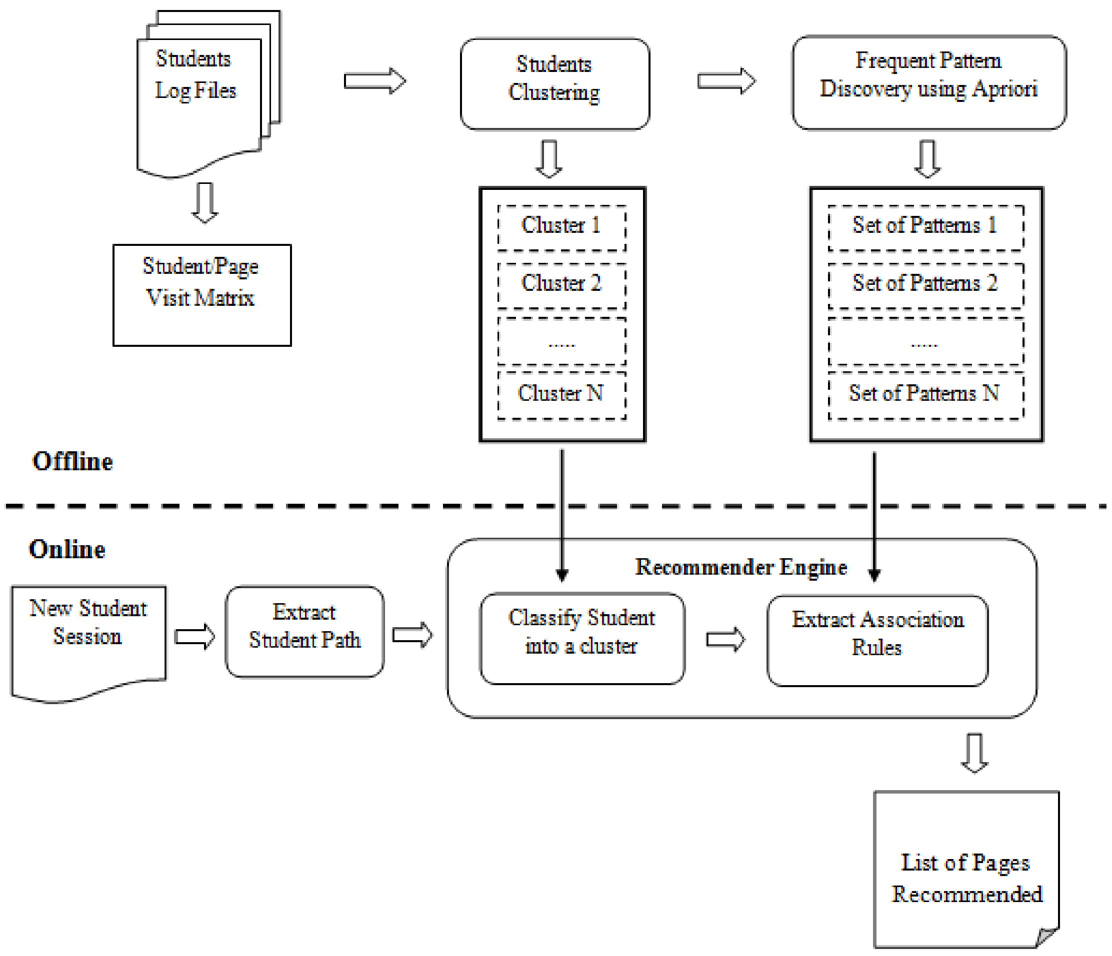


Figure 3: Proposed Link Recommendation Technique

This technique includes the three phases of web usage mining mentioned in the literature. Student/Page visit matrix is resulted from the pre-processing step, set of patterns will be discovered next, and finally the link recommendation process will be conducted online as described in the online section from figure 3.

Combinations of data mining techniques are applied in the proposed technique as described in figure 3. Clustering is used to group similar students according to their navigational paths, Apriori algorithm is used in order to discover frequent patterns, and classification and association rules are applied in the online phase to accomplish the link recommendation process.

The next sections in this chapter will provide detailed description of the proposed technique steps with explanation of algorithms and methods used in each step.

3.2. Data Pre-processing

It is necessary for web log files to be prepared and pre-processed in order to be used in the consequent phases of links recommendation. The raw data from log files cannot be used directly for analysis; it has to be cleansed in order to be analyzed. Data pre-processing can be decompose into two main steps with subtasks contained in each step as follows.

3.2.1. Web Log File Cleansing

Log files cleansing consist of removing useless requests from log files. We need to eliminate the irrelevant data in the log files. Usually requests concerning non-analyzed resources are removed. For example, requests for any file which might be included into a web page like graphical content of a web page. By cleansing log files obtained from users' interacting with websites, we can reduce the log files size to half of its original size and facilitate our upcoming tasks that require reading these log files. The log file cleansing phase consists of the following two tasks:

1. Attribute Filtering

Each record in the log files included the following attributes: date and time, IP address, URL method, the request status, the protocol status, the bytes sent, the protocol version, the user agent, and the referrer as demonstrated in table 1.

Table 1: Sample of Data before pre-processing

1.	2010-03-04 20:14:04 10.1.1.2 - GET /img/icon_plus.gif - 304 165 HTTP/1.0 Mozilla/4.0+(compatible;+MSIE+8.0;+Windows+NT+6.1;+WOW64;+Trident/4.0;+SLCC2;+.NET+CLR+2.0.50727;+.NET+CLR+3.5.30729;+.NET+CLR+3.0.30729;+Media+Center+PC+6.0;+InfoPath.2) http://10.248.127.19/
2.	2010-03-21 19:53:08 10.248.121.42 - GET /chap09_03.html - 304 141 HTTP/1.1 Mozilla/4.0+(compatible;+MSIE+6.0;+Windows+NT+5.1;+SV1;+.NET+CLR+1.1.4322;+.NET+CLR+2.0.50727;+.NET+CLR+3.0.04506.648;+.NET+CLR+3.5.21022) http://10.248.127.19/wrapnt_create_your_own_objects94.html
3.	2010-03-29 17:37:26 10.248.121.172 - GET /favicon.ico - 404 4203 HTTP/1.1 Mozilla/4.0+(compatible;+MSIE+8.0;+Windows+NT+5.1;+Trident/4.0;+.NET+CLR+1.1.4322;+.NET+CLR+2.0.50727;+.NET+CLR+3.0.4506.2152;+.NET+CLR+3.5.30729) -

Unwanted attributes were removed, such as HTTP version, status code, referrer, the protocol version, the user agent, and only the needed attributes were left like: Date/Time, IP address, and referrer. Table 2 shows partial set of the data after pre-processing and cleaning.

Table 2: Sample of Data after pre-processing

	Time	IP address	Date	Referrer
1.	20:14:04	10.1.1.2	2010-03-04	/img/icon_plus.gif
2.	19:53:08	10.248.121.42	2010-03-21	/chap09_03.html
3.	17:37:26	10.248.121.172	2010-03-29	/favicon.ico

2. Record Filtering

Record filtering process was done by checking the suffix of URL. All of .gif, .jpeg, .png, etc. were considered to be irrelevant and had to be removed except for HTML, and HTM which were related to the chapters' pages in the website. This process was performed by reading each record resulted from the attribute filtering and then the extension of the referrer was checked if it contains HTM or not. If it does contain then the record was kept, if not the record was removed.

3.2.2. Identification

In the second step of data pre-processing, several items needed to be grouped like users, sessions and pages. At the end of this step the log file will be a set of user/visit page transactions.

1. Users Identification

The user is identified by the computer IP address, but many users might use the same computer; which indicates that different users may have the same IP address. At first, the unique IP addresses that exist in the log files were identified. The purpose for these addresses was to define unique users. The process searches for unique IP addresses through the log file, whenever a new IP address was found the process added it to a list with new user ID. A list of unique users with their IDs and IP addresses was created. Table 3 shows sample of users' IDs identified in this phase.

Table 3: Sample of Unique Index of Users

User ID	User IP address	User ID	User IP address
U0	10.1.1.2	U5	10.248.120.70
U1	10.249.112.24	U6	10.248.127.16
U2	10.248.121.42	U7	10.248.120.76
U3	10.248.121.161	U8	10.248.121.213
U4	10.248.121.160	U9	10.248.120.138

2. Session Identification

As mentioned in user identification phase, the users who used the tutorial could use the same computer which means that they have the same IP address; also the users may visit the tutorial from many computers. The IP address could not be used to identify unique users. Session is introduced as the unit of identification of unique users. A session consists of pages accessed by a user in a central amount of time. If the time between pages' request exceeded a certain limit then the user was staring a new session. In this research the session threshold was defined as a 45 minute visit as default timeout i.e. 2700 second.

After applying this step the data in the server log was transformed into a set of sessions. Each session had a sequence of visited pages with their access time. This information was then used for creating student/page visit matrix. Table 6 shows a sample from users' session.

Table 4: Sample of Users' Sessions

User ID	User IP	Session Period	User ID	User IP	Session Period
U0	10.1.1.2	0	U6	10.1.1.2	1481
U1	10.1.1.2	0	U7	10.1.1.2	2285
U2	10.1.1.2	1359	U8	10.1.1.2	369
U3	10.1.1.2	2573	U9	10.1.1.2	2446
U4	10.1.1.2	385	U10	10.1.1.2	2290
U5	10.1.1.2	2571	U11	10.1.1.2	0

3. Page Identification

In this section a unique list of pages with their IDs was created. The pages were recognized by the log file, whenever the user has visited a page and the page was not in the list of the identified pages, this process added the page to the list of pages identified by giving it a new page ID.

The page list contained 195 pages; a sample from the pages' index is shown in table 5, and the full pages' list is shown in appendix A. The IDs starts from 1001 to 1195.

Table 5: Sample of Page Index

Page ID	Link	Number of words	Description of the page
P1001	/chap07_05.html	247	Reading Documentation
P1001	/chap08_11.html	275	Garbage collection
P1002	/chap03_03.html	429	Math methods
P1003	/chap05_04.html	504	Overloading
P1004	/chap02_01.html	480	More printing

Student/Page visit matrix was built after finishing data pre-processing phase, Table 6 shows a sample from the student/page visit matrix, A bigger sample is displayed in appendix B. The first column represents the users' ID, and the first row represents the pages' ID. The matrix is binary matrix where 1 represents that a student visited a page and 0 represents that a student did not visit a page.

Table 6: Sample of Student/Page visit Matrix

User ID	p1	p2	p3	p4	p5	p6	p7	p8
10001	0	0	0	0	0	0	0	0
10002	1	0	0	0	0	0	0	0
10003	0	0	0	0	0	0	0	0
10004	0	0	0	0	1	0	0	0
10005	0	0	1	1	0	0	1	1
10006	0	0	0	0	1	0	0	0
10007	0	0	0	0	0	0	0	0
10008	0	0	0	0	0	0	0	0

3.3. Students Clustering

Data clustering is implemented in the proposed technique of links recommendation. In this research clustering is applied on a set of students in order to find similar groups of students according to their navigational paths based on student/page visit matrix resulted from the data pre-processing step. The navigational paths for groups of students resulted from the clustering process are extracted and used as input for pattern discovery process, where Apriori algorithm is applied on each group of students paths in order to discover related patterns. The main objective of data clustering in this technique is to group similar students in order to efficiently discover larger number of related patterns for each cluster of students.

K-means clustering is one of the oldest unsupervised clustering algorithms. This algorithm is used in the proposed technique for link recommendation since it is one of the simplest and most widely used clustering algorithms.

Proximity measures (distance functions) are used according to the type of data being clustered. Cosine distance function is the function applied in k-means clustering process in this research. This distance function considers each row in the student/page visit matrix as a vector and measures the cosine of the angle between each vector and the cluster centroid.

The validity of clusters obtained from clustering operation can be measured using different validity measures. In this research, the appropriate number of clusters (*k*) is determined by evaluating the clusters obtained using two criteria:

- Horizontal Silhouette: measures the healthiness of clusters in respect to the correct allocation of points into clusters.

- Vertical Silhouette: measures the fair distribution of students to the clusters obtained, i.e. the uniformity in the number of students assigned to each cluster. The difference between ratio distributions of students to clusters is computed in order to choose the appropriate number of clusters applied in the proposed technique.

These two criteria affect the number of patterns discovered in each cluster and the degree of correlation between pages of these patterns.

3.4. Frequent Pattern Discovery Using Apriori Algorithm

Discovering frequent navigational patterns is implemented in the proposed technique in order to infer association rules from these patterns, and then recommend links to students. Apriori algorithm is used in this thesis to discover frequent patterns of pages.

In this thesis, frequent patterns of pages, visited by a percentage of the overall students no lower than specific support percentage, are discovered. Consider the database containing students' navigational paths in table 7 and assume that the support threshold is 50%. Sets of frequent patterns of different lengths that satisfy the support threshold are shown in table 7(a), table 7(b), and table 7(c).

Each set of pages that has been visited by 2 or more students will satisfy the support percentage, since the total number of students is 4 and that will result in 50% and larger support percentage for each set of pages.

Table 7: Database

Student ID	Path
1	P1,P3,P4
2	P2,P3,P5
3	P1,P2,P3,P5
4	P2,P5

Table 7(a): Frequent Patterns of Length=1

Frequent Pattern	Student Count
{P1}	2
{P2}	3
{P3}	3
{P5}	3

Table 7(b): Frequent Patterns of Length=2

Frequent Pattern	Student Count
{P1,P3}	2
{P2,P3}	2
{P2,P5}	3
{P3,P5}	2

Table 7(c): Frequent Patterns of Length=3

Frequent Pattern	Student Count
{P2,P3,P5}	2

The support of a frequent pattern is commonly expressed as a percentage. For example, a pattern is considered to be frequent if it has a support percentage equal to or greater than 50%. This means that a pattern must be seen in half of the records included in the database. In this research, a frequent pattern is a set of pages accessed by a number of students who satisfy a specific percentage of the overall students within the same cluster of students.

The support percentage S_p is the percentage of number of students who visited a specific set of pages to the total number of students. And it is calculated as shown in equation (1).

$$S_p = \frac{S_c\ (F1)}{S} \ * 100\ \% \qquad \text{.....................................} \quad (1)$$

Where *F1* is a set of pages (pattern), $S_c\ (F1)$ is the number of students who visited $F1$, and S is the total number of students visited the website

3.4.1. Apriori Algorithm

Frequent pattern mining was first introduced by (Agrawal et. al., 1993) for market basket analysis. They defined a property called Apriori which articulate that *"All subsets of a frequent itemset must also be frequent"*. In other words an itemset is frequent only if all its subitems are frequent. This property gives an indication of how Apriori algorithm works. It implies that frequent itemsets are discovered by first scanning the database to find frequent 1-itemsets (items of length 1), then using the itemsets of length 1 to generate frequent 2-itemsets, and so one. This process iterates until no more frequent *k*-itemsets can be discovered, where *k*-itemsets are all itemsets (patterns) of length *k*

The Apriori algorithm is an iterative approach which is illustrated in Figure 4.

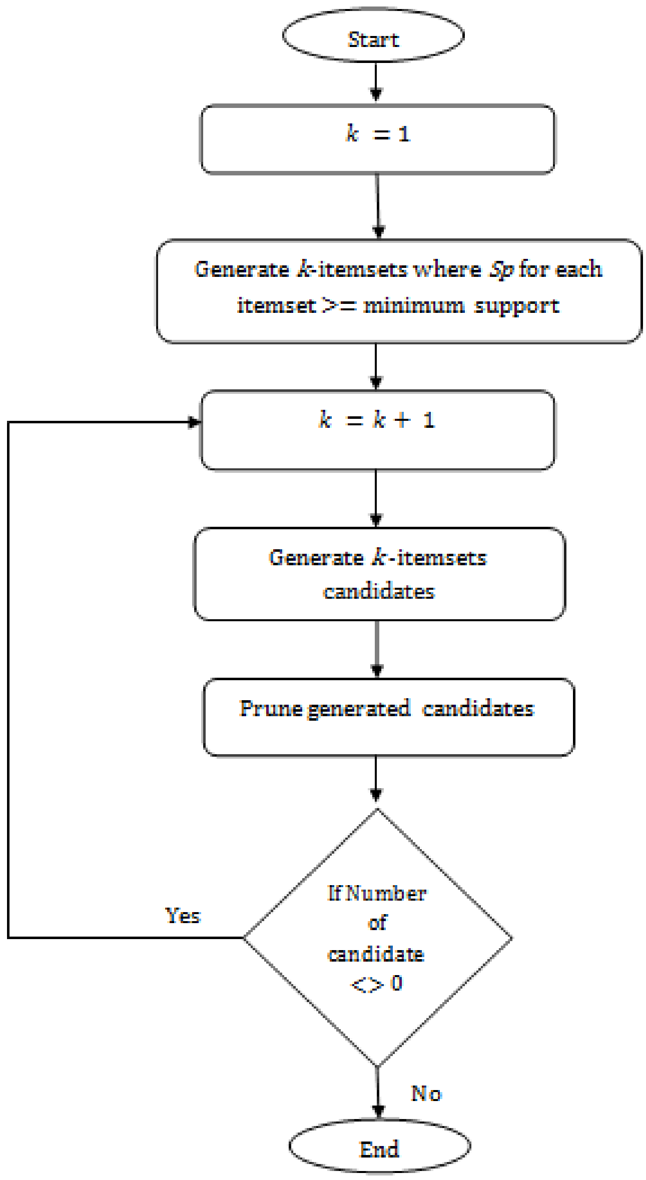


Figure 4: Aprior Algorithm Steps

In the first step in Apriori Algorithm, a set of items of length 1 (*k*-itemset when *k*=1) are found by scanning the database containing students' transactions. For each frequent pattern, the support percentage is calculated and items that do not satisfy pre-defined support threshold are eliminated in the pruning step.

The 1-itemsets resulted from the pruning step are used as input to the following steps: Candidate generation and candidate pruning.

- Candidate Generation.

  In this step, the Apriori property is utilized. The property is based on the following observation, if an itemset *fp* is not frequent or in other words, do not satisfy a support threshold, and if item *p* is added to the itemset *fp*, then the resulting itemset cannot occur more frequently than *fp*.

  Based on that, candidate generation is performed using the frequent patterns of smaller length to generate longer frequent patterns that are also frequent. *k+1*-itemsets are generated using *k*-itemsets. This is accomplished by joining the *k*-itemsets with themselves.

  The join step is performed between *k*-itemset and itself, the itemsets are of the same length *k*. if their first *k-1* items are in common. For example, if {p1,p2,p3} is joined with {p2,p1,p4}, it will result in generating a new itemset of length 4 which is {p1,p2,p3,p4}.

- Pruning the Candidates

  Candidate generation step results in all possible itemsets but not all of them are frequent. A scan on the database is required to determine whether each candidate satisfy the support percentage, and eliminate any itemset that do not satisfy the support percentage.

3.4.2. An Example of the Use of Apriori Algorithm

An example explains the use of Apriori algorithm for discovering frequent patterns in a database that records students' navigation through a website is illustrated. The database contains two columns representing, student ID and a sequence of pages visited by each student. Table 8 shows the database that will be used in the example.

Table 8: Students' Navigation Database

Student ID	Path
1	P1,P2,P5
2	P2,P4
3	P2,P3,P6
4	P1,P2,P4
5	P1,P3
6	P2,P3
7	P1,P3
8	P1,P2,P3,P5
9	P1,P2,P3
10	P1,P2

In the first step, a scan on the database is performed in order to find all possible 1-itemsets and count their number of occurrences. Table 9 shows 1-itemsets found in this example.

Table 9: 1-Itemsets Found in the Database

1-itemset	Count
{P1}	7
{P2}	8
{P3}	6
{P4}	2
{P5}	2
{P6}	1

A support percentage of 20% is determined to consider a pattern to be frequent in this example. The 1-itemsets in table 9 will be pruned to only the itemsets that satisfy the support percentage condition. The pruned 1-itemsets are shown in table 10.

Table 10: Pruned 1-Itemsets

1-itemset	Count
{P1}	7
{P2}	8
{P3}	6
{P4}	2
{P5}	2

After having found the frequent 1-itemsets, candidate generation step is performed to find all possible 2-itemsets from 1-itemsets in table 10. The candidates 2-itemsets generated are shown in table 11.

Table 11: 2-Itemsets Generated from the Database

2-itemset	Count
{P1,P2}	5
{P1,P3}	4
{P1,P4}	1
{P1,P5}	2
{P2,P3}	4
{P2,P4}	2
{P2,P5}	2
{P3,P4}	0
{P3,P5}	1
{P4,P5}	0

These items are then pruned to satisfy the 20% support percentage based on the database records shown in table 8. The pruned 2-itemsets are shown in table 12.

Table 12: Pruned 2-Itemsets

2-itemset	Count
{P1,P2}	5
{P1,P3}	4
{P1,P5}	2
{P2,P3}	4
{P2,P4}	2
{P2,P5}	2

Next, 3-itemsets are generated by joining 2-itemsets, shown in table 12 with itself. The result is shown in table 13.

Table 13: 3-Itemsets Generated from the Database

3-itemset	Count
{P1,P2,P3}	2
{P1,P2,P5}	2
{P2,P3,P4}	0
{P2,P3,P5}	1

As shown in the previous step, these items are then pruned to satisfy the 20% support percentage defined. The pruned 3-itemsets are shown in table 14.

Table 14: Pruned 3-Itemsets

3-itemset	Count
{P1,P2,P3}	2
{P1,P2,P5}	2

Since the number of candidates after pruning is not equal to zero, another iteration is needed to find 4-itemsets. By joining 3-itemsets shown in table 14 with themselves, it will result in a single itemset {P1,P2,P3,P5}. However this itemset has occurred only once in the database, which indicates that it does not satisfy the 20% support percentage defined. After pruning the 4-itemset we will have zero candidates, and this will terminate the algorithm.

The set of maximal frequent patterns found in this example are shown in table 15.

Table 15: Maximal Discovered Frequent Patterns

Frequent Patterns	Count
{P2,P4}	2
{P1,P2,P3}	2
{P1,P2,P5}	2

3.5. Recommender Engine

3.5.1. Classifying a Student into a Cluster

In this research, link recommendations are generated for new students. In order to accomplish this task the new students must be classified into one of the clusters first according the navigational path of that new student. A similarity measure is needed to determine the most suitable cluster for a student, in order to use the frequent patterns generated from that cluster.

Since we are dealing with binary data that represent student visit/ no visit to pages, then the most appropriate similarity measure in our case is the Jaccard coefficient. The Jaccard similarity coefficient is a number between 0 and 1, where 1 means that both students are exactly the same and 0 means they are absolutely different.

Table 16 represents a sample of the student/page visit matrix that is used in this research.

Table 16: Sample of Student/Page Visit Matrix

Student ID	P1	P2	P3	P4	P5
S1	1	1	1	0	0
S2	1	1	0	0	1
S3	0	0	1	1	0

The similarity between two students S1 and S2 is Sm(S1,S2) calculated using equation (2).

$$S_m\ (S_1, S_2) = \frac{a}{a+b+c} \qquad \text{.....................................} \quad (2)$$

Where a is the number of pages visited by both $S_1$and S_2, b is the number of pages visited by $S_1$and not visited by S_2, and c is the number of pages visited by $S_2$and not visited by S_1 .

Thus, $S_m\,(S_1, S_2) = \frac{2}{2+1+1} = 0.5$.

In the proposed technique, we classify a student into a cluster by computing the average Jaccard similarity between that student and every student in each cluster as shown in table 17. This will result in finding the average of similarity between a student and the clusters found.

Table 17: A Student Average Jaccard Similarity for each Cluster

Student ID	Cluster ID	Average Jaccard Similarity
1	1	0.0176
1	2	0.2900
1	3	0.0740
1	4	0.0175
1	5	0.0274

The student will belong to the cluster that has the maximum average similarity. In table 16, student with ID equal to 1 belongs to cluster 2 since it has the maximum average jaccard similarity result.

3.5.2. Extracting Association Rules

Association rules are used in the online recommender engine. Discovering association rules in general consists of two phases, which are:

- Discover all frequent itemsets: in the proposed technique, this phase is applied offline using Apriori algorithm as explained in section 3.4.
- Generate association rules from the frequent itemsets and calculate the confidence for each rule as follows:
    1. For each frequent itemset f of length *k*, generate all possible combinations of items.

2. For every combination of items $Nk=\{C_1, C_2,\dots, C_k\}$, output the rule R_k.

$$R_k = \{ C_1, C_2,\dots, C_{k-1} \} \rightarrow \{C_k\}$$

The confidence of the rule R_k is $Conf(R_k)$ calculated using equation (3).

$$Conf(R_k) = \frac{S_c(Nk)}{S_c\,(Nk - C_k)} * 100\,\% \qquad \text{..............................} \qquad (3)$$

Where $S_c(Nk)$ is the count of students who visited set of pagesNk, and $S_c\,(Nk - C_k)$ is the count of students who visited subset of Nk which is $= \{ C_1, C_2,\dots, C_{k-1} \}$.

For example, if we have frequent pattern *f={p1,p2,p3}*,then all possible combinations of itemsets, association rules extracted from these itemsets , and the confidence of each rule are shown in table 18. The confidence is calculated using the count values in table 12 and table 14.

Table 18: Possible Rules Discovered from *f*

Combinations of an itemset	Rules	Confidence
{ P1,P2,P3}	P1,P2 → P3	40%
{ P1,P3,P2}	P1,P3 → P2	50%
{ P2,P3,P1}	P2,P3 → P1	50%

4. Experiments and Results

This chapter illustrates the results obtained from several experiments which has been carried out. These experiments were conducted to test the link recommendation technique proposed in this thesis. In addition, a comparison of the results obtained from applying this technique to recommend links is introduced.

The data used in these experiments are collected from log files for web-based java tutorial uploaded on a server and accessed by second year students, who are studying Object Oriented Programming (OOP) courses at King Abdullah II School for Information Technology (KASIT) at Jordan University (JU).

OOP is taught at our faculty as two separate courses, OOP1 and OOP2. Both courses are mandatory requirement for Computer Information Systems (CIS) bachelor students in order to complete their second year successfully.

The log files acquired were pre-processed and cleansed for research purposes by other master students at the faculty. 7853 records that represent 438 students accessing 195 hypermedia pages have been obtained.

As explained in figure 3, the technique for link recommendation proposed in this thesis consists of two phases which includes (i) an offline phase which identifies the frequent patterns in students' navigation, and (ii) an online phase which recommends links to new students while they navigate through the hypermedia.

The data collected was divided for two disjoint data sets in order to test the proposed technique. 90% of the data was used as the training data in the offline phase, and the remaining 10% of data was used as the testing data in the online phase.

Table 19 outlines the number of students and pages accessed by the students in each data set, after applying random sampling on the student/page access matrix.

Table 19: Training and Testing Datasets

Data set	Number of students	Total number of pages accessed
Training data	394	7000
Testing data	44	853

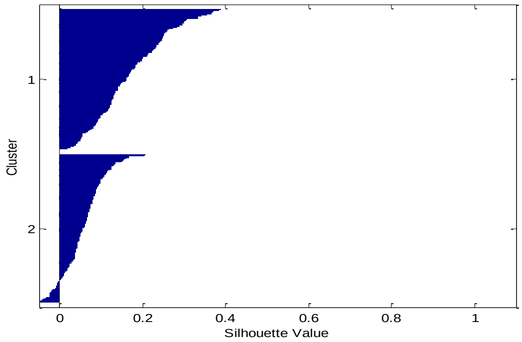
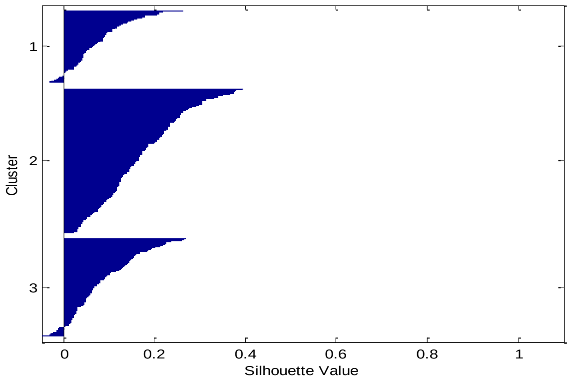
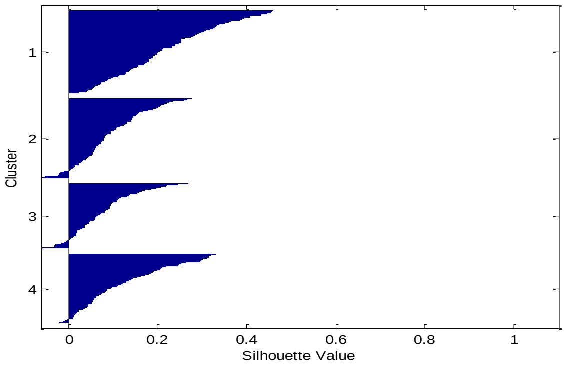
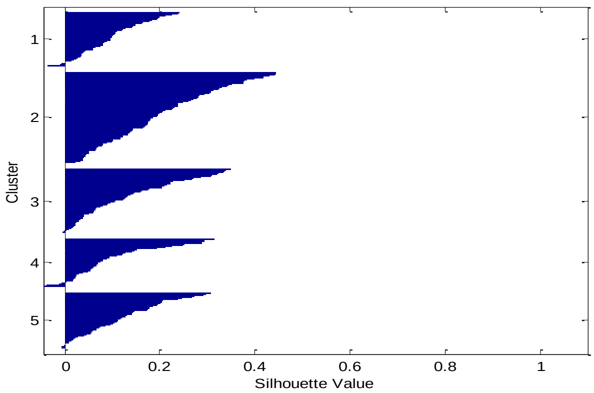
4.1. Clustering of Training Students

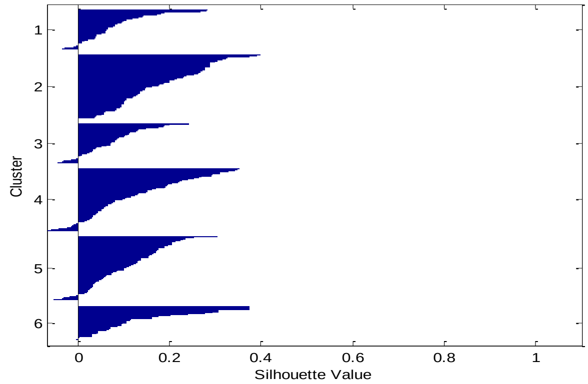
In the first experiment, *k*-means algorithm has been executed with varying number of clusters (*k*) in order to find out the appropriate number of existing clusters in the training data.

The silhouette methods are used to measure the goodness of clusters obtained after performing *k*-means clustering with varying *k* value from 2 to 6. Results of the silhouette methods were used to help in deciding the appropriate *k* value for the training data. It is important to have uniform clusters, similar in the number of students in each cluster and the total number of pages accessed by these students because it affects the number of frequent patterns discovered in each cluster

Table 20 shows the Horizontal silhouette plots generated for every *k*-means clustering experiment with different *k* value varying from 2 to 6.

Table 20: Horizontal Silhouette Plots for Clustering with Varying *k* Value

K	**Silhouette plot**
K=2	
K=3	
K=4	
K=5	

K=6	

The silhouette plots generated in table 20 gives a good indication of the healthiness of clusters in respect to the correct allocation of students into clusters. In this research the validity of clusters obtained also depends on the fair distribution of students in each cluster and the total number of pages accessed by each cluster of students.

Table 21 demonstrates the healthiness of each cluster depending on two silhouette criteria and provide details about number of students and the total pages accessed by those students in each cluster. The silhouette values are analysed in both horizontal and vertical directions where silhouette (average horizontal) measures the degree of separation of each cluster from other clusters, and silhouette (average vertical) measures the distribution of students in all clusters.

Table 21: Clustering Analysis

K	**Silhouette (Average Horizontal)**	**Silhouette (Average Vertical)**	**Number of Students/ Number of Pages**					
K=2	0.112	0.025	cluster 1 192/3170	cluster 2 202/3830				
K=3	0.124	0.08	cluster 1 90/1595	cluster 2 181/3075	cluster 3 123/2330			
K=4	0.131	0.025	cluster 1 110/1547	cluster 2 106/2018	cluster 3 86/1566	cluster 4 92/1869		
K=5	0.140	0.06	cluster 1 68/1208	cluster 2 114/1609	cluster 3 81/1534	cluster 4 60/1283	cluster 5 71/1366	
K=6	0.123	0.57	cluster 1 52/927	cluster 2 82/1336	cluster 3 51/964	cluster 4 81/1802	cluster 5 83/1542	cluster 6 45/429

Table 21 shows that when the number of clusters increases then the total number of pages accessed by students of each cluster is not balanced between clusters, (for example, the total number of pages accessed in cluster 6 and cluster 5 when *k=6)*. Notice that the distribution of students to clusters (vertical silhouette) when *k=6* is the largest of all values. Also notice that the data are more balanced when smaller number of clusters *k=4* and *k=5* is used, and the average horizontal silhouette measure confirms these observations.

The problem with having large number of clusters is the number of frequent patterns that can be obtained from students' navigation through the website, in other words the number of pages visited in each cluster affects the chances of getting frequent patterns with high support threshold. The next experiment is conducted to choose the best number of clusters between *k=4* and *k=5* clustering.

4.2. Frequent Pattern Discovery

For the purpose of understanding the obtained experiments and results conducted. Two terms were used and they must be differentiated which are, frequent pattern and maximal frequent pattern. Maximal frequent pattern is a pattern that is not a subset of any other frequent pattern (Gouda, et al., 2005). Frequent pattern term refers to any itemset that is frequent without taking into consideration if it is a subset of any larger frequent pattern. Frequent patterns were used in this research to generate recommendation links to students.

In this experiment, frequent patterns are discovered in students' navigation using Apriori algorithm. A comparison between frequent patterns discovered in clusters when *K=4* and *k=5* is performed. This comparison is conducted with varying support percentage from low to high and check the uniformity in the number of frequent patterns discovered in each cluster in order to decide the most appropriate k value to be used for this data set.

Clustering with $k=4$ and $k=5$, as shown in table 21 has the largest average horizontal and vertical silhouette. Table 22 and table 23 show the results of applying Apriori algorithm on the two sets of clusters where $k=4$ and $k=5$ and the number of maximal frequent patterns obtained in each cluster and their average support with varying support threshold.

Table 22: Frequent Patterns Discovery Using Apriori with Varying Average Supports (K=4)

	Support percentage = 15%		**Support Percentage =20%**		**Support Percentage =25%**		**Support Percentage =30%**	
	Number of frequent patterns	Average support percentage	Number of frequent patterns	Average support percentage	Number of frequent patterns	Average support percentage	Number of frequent patterns	Average support percentage
Cluster 1	403	15.5%	204	20.1%	73	25.8%	30	30.8%
Cluster 2	65	15.2%	16	21.1%	4	25.9%	0	0%
Cluster 3	22	15.3%	1	20.9%	0	0%	0	0%
Cluster 4	899	15.2%	319	20.8%	83	25.4%	0	0%

Table 23: Frequent Patterns Discovery Using Apriori with Varying Average Supports (K=5)

	Support percentage = 15%		**Support Percentage =20%**		**Support Percentage =25%**		**Support Percentage =30%**	
	Number of frequent patterns	Average support percentage	Number of frequent patterns	Average support percentage	Number of frequent patterns	Average support percentage	Number of frequent patterns	Average support percentage
Cluster 1	38	16.2%	9	21.0%	0	0%	0	0%
Cluster 2	362	15.8%	162	20.3%	69	25.9%	25	31.6%
Cluster 3	747	16.0%	350	21.0%	97	26.4%	5	30.8%
Cluster 4	190	15.0%	82	20.1%	37	25.1%	5	30.3%
Cluster 5	107	15.5%	45	21.2%	12	25.4%	11	31.4%

It is noticed that even though clusters obtained when *k=4* have large number of pages accessed this does not imply having large numbers of frequent patterns discovered in each cluster. The large number of students in each cluster increases the number of students that must visit some pattern so that it can be considered to be frequent according to a specific minimum support threshold. Also, it can be notice that the number of frequent patterns in clusters where *k=5* is more uniform even with high support threshold compared with frequent patterns discovered when *k=4*. Also the average support of the patterns in clusters where *k=5* is larger than the average support of frequent patterns in clusters where *k=4.*

From both experiments conducted in section 4.1 and section 4.2 clustering with *k* value equal to 5 is considered to be the most appropriate clustering performed on this data set of students.

4.3. Classification of Test Students

Recommending links to a student requires classification of that student into a cluster first. Then the frequent patterns discovered from that cluster are invoked in order to recommend links to that student.

As explained in section 3.5. The average Jaccard similarity coefficient is computed to find the similarity between each student and the five obtained clusters in order to classify a student into the cluster that has the maximum average Jaccard similarity coefficient. Table 24 demonstrates the results of classifying the 44 students in the test dataset to the 5 clusters obtained from *k*-means clustering operation.

Table 24: **Results of Classifying Test Students into Clusters Based on Jaccard Similarity Coefficient (k=5)**

Student ID	Cluster ID	Jaccard similarity coefficient	Student total pages visited
1	2	0.2062	25
2	5	0.1073	16
3	5	0.1503	8
4	1	0.0405	6
5	5	0.0634	7
6	3	0.1675	10
7	3	0.2387	23
8	2	0.1928	25
9	5	0.0485	8
10	2	0.0790	8
11	2	0.2998	10
12	2	0.2588	35
13	4	0.1601	76
14	4	0.0306	4
15	1	0.0485	7
16	1	0.0410	6
17	2	0.1579	12
18	2	0.1969	11
19	4	0.2197	23
20	5	0.0973	14
21	2	0.2890	16
22	2	0.0784	2
23	3	0.1591	10
24	4	0.0797	27
25	5	0.0554	11
26	2	0.0903	56
27	5	0.1165	17
28	3	0.0975	25
29	5	0.1492	47
30	5	0.1285	44
31	1	0.0688	12
32	1	0.0705	7
33	1	0.1217	30
34	2	0.2560	13
35	3	0.2608	21
36	2	0.0765	9
37	3	0.1208	7
38	3	0.1360	8
39	1	0.0570	7
40	2	0.1343	40
41	4	0.1554	58
42	5	0.0534	28
43	2	0.2134	21
44	4	0.0204	3

Table 24 demonstrates the cluster ID that each student belongs to after computing the Jaccard similarity coefficient between each student the 5 obtained clusters. It can be notice that students who have small number of pages in their navigational path also have low Jaccard similarity value since it is hard to determine if a student is similar to a cluster of students from small number of pages compared.

Table 25 shows the distribution of the test student dataset classified into 5 clusters.

Table 25: Test Students Dataset Classified into 5 Clusters

Cluster number	Number of students
Cluster 1	7
Cluster 2	14
Cluster 3	7
Cluster 4	6
Cluster 5	10
Total number of Students	**44**

4.4. Links Recommendation for Sample of Students

In this experiment, sample of test students and their recommended links is outlined. The frequent patterns discovered in the 5 clusters with support percentage equal to 15% are applied in this experiment. Sample of 5 students from different clusters is taken to demonstrate the links recommended for each student using the frequent patterns discovered in each cluster. The results of the experiments applied on 5 students shows names of links recommended to each student, these links are ordered in descending order according to the confidence value of each rule.

For example, student with *ID = 33* have been classified into cluster 1. The frequent patterns discovered in cluster 1 are used to recommend links to that student. Table 26 shows the links recommended to the student with *ID=33* after visiting 3 pages.

Table 26: Links Recommendation and their Confidence for Student with ID=33

Student path	Link recommended	Confidence
Building trees , Expression trees , Traversal →	Traversing trees	100 %
Building trees , Expression trees , Traversal →	A tree node	92.3%
Building trees , Expression trees , Traversal →	Array implementation of trees	84.6%
Building trees , Expression trees , Traversal →	Defining an abstract class	84.6%

It can be seen in table 26 that the pages recommended to this students are all very related to the topic of pages he visited. For example, after visiting building trees, expression trees, and traversal, the pages traversing trees, tree node, and array implementation of trees are recommended. These pages are recommended in order to complete the intended learning goal of the student which can be here trees in Java. These links recommended have high confidence values according to other students' navigation.

Student with *ID = 43* have been classified into cluster 2. The frequent patterns discovered in cluster 2 are used to recommend links to that student. Table 27 shows the links recommended to the student with *ID=43* after visiting 3 pages.

Table 27: Links Recommendation and their Confidence for Student with ID=43

Student path	Link recommended	Confidence
Variables , Assignment , Printing variables →	More printing	89.7%
Variables , Assignment , Printing variables →	Operators	84.6%
Variables , Assignment , Printing variables →	Keywords	82.1%
Variables , Assignment , Printing variables →	Composition	79.5%
Variables , Assignment , Printing variables →	Formal and natural languages	74.4%
Variables , Assignment , Printing variables →	Operators for Strings	71.8%
Variables , Assignment , Printing variables →	Floating point	66.7%
Variables , Assignment , Printing variables →	The first program	66.7%
Variables , Assignment , Printing variables →	Order of operations	66.7%
Variables , Assignment , Printing variables →	What is a programming language	64.1%

It can be noticed from table 27 that the pages recommended are not all directly related to the pages visited by the student but here the confidence values are low. It can be seen that some pages recommended like more printing, operators, and keywords are very related to the pages visited by the student, also they will help him complete the intended learning goal which can be here introduction to Java.

Student with *ID = 37* have been classified into cluster 3. The frequent patterns discovered in cluster 3 are used to recommend links to that student. Table 28 shows the links recommended to the student with *ID=37* after visiting 3 pages.

Table 28: Links Recommendation and their Confidence for Student with ID=37

Student path	Link recommended	Confidence
Math methods , Composition , Classes and methods ➔	Adding new methods	82.4%
Math methods , Composition , Classes and methods ➔	Methods with results	76.5%
Math methods , Composition , Classes and methods ➔	Alternative execution	76.5%
Math methods , Composition , Classes and methods ➔	Chained conditionals	76.5%
Math methods , Composition , Classes and methods ➔	The return statement	76.5%

It can be noticed from table 28 that the pages recommended for this student are related to pages visited by that student. The page with higher confidence, add new method is very related to the pages visited.

Student with *ID = 24* have been classified into cluster 4. The frequent patterns discovered in cluster 4 are used to recommend links to that student. Table 29 shows the links recommended to the student with *ID=24* after visiting 3 pages.

Table 29: Links Recommendation and their Confidence for Student with ID=24

Student path	Link recommended	Confidence
Printing an object , Creating a new object , Objects as parameters ➔	Packages	90%
Printing an object , Creating a new object , Objects as parameters ➔	Instance variables	90%
Printing an object , Creating a new object , Objects as parameters ➔	Objects are mutable	90%
Printing an object , Creating a new object , Objects as parameters ➔	Aliasing	90%
Printing an object , Creating a new object , Objects as parameters ➔	Time	90%
Printing an object , Creating a new object , Objects as parameters ➔	Constructors	90%
Printing an object , Creating a new object , Objects as parameters ➔	Pure functions	90%
Printing an object , Creating a new object , Objects as parameters ➔	What's interesting	90%

It can be seen in table 29 that pages recommended to the student after visiting the pages, printing an object, create new object, and object as parameters are all related to the topic of previously visited pages, for example constructors, instance variables and packages.

Student with *ID = 3* have been classified into cluster 5. The frequent patterns discovered in cluster 5 are used to recommend links to that student. Table 30 shows the links recommended to the student with *ID=3* after visiting 3 pages.

Table 30: Links Recommendation and their Confidence for Student with ID=3

Student path	Link recommended	Confidence
Arrays and objects , for loops , Random numbers →	Array length	95.2%
Arrays and objects , for loops , Random numbers →	Array of random numbers	90.5%
Arrays and objects , for loops , Random numbers →	Copying arrays	90.5%
Arrays and objects , for loops , Random numbers →	Accessing elements	85.7%
Arrays and objects , for loops , Random numbers →	Counting	76.2%
Arrays and objects , for loops , Random numbers →	Statistics	71.4%
Arrays and objects , for loops , Random numbers →	A single-pass solution	66.7%
Arrays and objects , for loops , Random numbers →	Many buckets	57.1%
Arrays and objects , for loops , Random numbers →	Arrays of cards	52.4%

It can be noticed from table 30 that the recommended pages are related to the topic of pages visited by the student. The pages recommended that has highest confidence values are more logically related to the topic of the visited pages than others. For example, array length, array of random number, and copying arrays pages are more logically related to the pages already visited by the student than other pages recommended.

From the experiments applied on a sample of 5 students, it can be perceived that all recommended pages are related to the pages already visited by the student. However the top 50% of the recommended pages are logically related to the subject of the pages visited by the student more than the bottom 50% of the pages recommended.

4.5. Validation of Links Recommendation Results

Since each student has many learning objectives or tasks to achieve when navigating through a hypermedia. Each page visited by a student will help him in understanding or achieving his task. Several experiments are conducted to validate the correctness of links recommended to each student in the test dataset by comparing them with the student actual navigational path and check if the recommended links have been visited while the students navigates the website

The primary aim of this experiment is to figure out the most appropriate number of pages visited (*P)* by a student in order to start recommending a page link. To figure out the most appropriate value for *P* several experiments are conducted with varying *P* value before links recommendation is made. The links recommended for a student are validated by comparing them with the actual pages visited by that student and calculate the percentage of actually visited pages from the number of pages recommended. Take into consideration that links recommended but not visited can be logically related to the student navigational path, but for some reasons like changing the learning goal of the student or the student stop navigating the hypermedia, these pages are not visited.

The following subsections analyze the results obtained from recommending links to students after 3 pages visited, then after 4 pages visited and so on until no recommendations can be made.

The number of students that has generated recommendation links decreases while the *P* value increases since the probability that a student visits for example 8 pages sequentially that are already been discovered as a pattern is low. Table 31 shows the number of students who have generated links recommendation with regarding to the number of pages visited.

Table 31: Number of Students with Generated links Recommendation after Varying Number of Initial Visited Pages

Number of initial visited pages (*P*)	Number of students with generated recommendations
3	31
4	21
5	15
6	11
7	8
8	6
9	2

4.5.1. Page Recommendations after Visiting 3 Initial Pages

Table 32 shows details about students who had recommendation links after visiting 3 initial pages. The numbers of links recommended to each student and the percentage of actually visited links from the recommended ones are listed in the table. A list of page IDs recommended to each student after visiting 3 pages is shown in appendix C.

Table 32: Page Recommendation after Visiting 3 pages

Student ID	Cluster ID	Total number of pages visited	Number of recommended links	Actually visited %	Not visited %
1	2	25	15	53%	47%
2	5	16	10	20%	80%
3	5	8	9	33%	67%
6	3	10	1	100%	0%
7	3	23	1	0%	100%
8	2	25	13	46%	54%
11	2	10	10	70%	30%
12	2	35	10	90%	10%
13	4	76	5	100%	0%
15	1	7	1	0%	100%
17	2	12	11	9%	91%
18	2	11	15	33%	67%
19	4	23	3	67%	33%
20	5	14	8	13%	88%
21	2	16	14	79%	21%
23	3	10	17	41%	59%
24	4	27	8	0%	100%
27	5	17	11	18%	82%

28	3	25	10	40%	60%
29	5	47	10	50%	50%
30	5	44	11	27%	73%
31	1	12	4	0%	100%
32	1	7	4	50%	50%
33	1	30	4	50%	50%
34	2	13	13	62%	38%
35	3	21	1	100%	0%
37	3	7	5	0%	100%
38	3	8	18	28%	72%
40	2	40	10	70%	30%
41	4	58	5	60%	40%
43	2	21	15	60%	40%
			Average	44%	56%

Actually visited links percentage is calculated by dividing the number of links recommended to a student and actually visited by that student on the total number of recommended links. The complement of the actually visited links percentage is considered to be not visited links percentage which represents the links recommended but not visited by a student.

It can be noticed from table 32 that some students have low actually visited pages percentage. This can be referred to one of the following reasons, firstly, the student navigational path is short, i.e. the student did not visit enough pages so that a pattern can be discovered in his behaviour. Secondly, the student navigational path is long enough but no frequent 3 pages pattern has been discovered which indicates that the student has been browsing the hypermedia randomly with no particular learning goal in mind.

Also, it can be seen that there are some student IDs missing between 1 and 44. The missing IDs belong to students who did not have any recommended links after 3 page clicks because these students do not fully belong to any cluster based on the Jaccard similarity results shown in table 24. The students have low average Jaccard similarity values which means they are not near to any cluster.

4.5.2. Page Recommendations after visiting 4 pages

The following experiment provides details about students who had recommendation links after visiting 4 initial pages. Table 33 shows student IDs who had links recommendation after visiting 4 initial pages, the number of links recommended to each student, and the percentage of actually visited links from the recommended ones. . A list of page IDs recommended to each student after visiting 4 pages is shown in appendix D.

Table 33: Page Recommendation after Visiting 4 Pages

Student ID	Cluster ID	Total number of pages visited	Number of recommended pages	Actually visited %	Not visited %
1	2	25	11	36%	64%
2	5	16	6	0%	100%
3	5	8	6	33%	67%
8	2	25	8	13%	88%
11	2	10	8	75%	25%
12	2	35	8	88%	13%
13	4	76	3	100%	0%
18	2	11	13	31%	69%
19	4	23	1	100%	0%
20	5	14	7	0%	100%
21	2	16	11	73%	27%
23	3	10	11	55%	45%
27	5	17	6	17%	83%
28	3	25	5	20%	80%
29	5	47	7	29%	71%
30	5	44	8	25%	75%
32	1	7	1	100%	0%
34	2	13	12	58%	42%
38	3	8	11	36%	64%
40	2	40	8	63%	38%
43	2	21	12	50%	50%
			Average	48%	52%

It can be noticed from table 33 that the number of students with generated recommended links decreased, and it will keep decreasing in the following experiments because of the Aprori property. The average of actually visited pages is near to the one in the previous experiment.

4.5.3. Page Recommendations after visiting 5 pages

The following experiment provides details about students who had recommendation links after visiting 5 pages. Table 34 shows the same details as the previous two experiments, the students IDs and number of links recommended to each student for students who had recommendations after visiting 5 initial pages. A list of page IDs recommended to each student after visiting 5 pages is shown in appendix E.

Table 34: Page Recommendation after Visiting 5 Pages

Student ID	Cluster ID	Total number of pages visited	Number of recommended pages	Actually visited %	Not visited %
1	2	25	10	30%	70%
3	5	8	4	0%	100%
8	2	25	4	0%	100%
11	2	10	7	71%	29%
12	2	35	7	86%	14%
18	2	11	10	10%	90%
21	2	16	10	70%	30%
23	3	10	5	80%	20%
27	5	17	4	0%	100%
29	5	47	6	17%	83%
30	5	44	5	20%	80%
34	2	13	10	60%	40%
38	3	8	8	38%	63%
40	2	40	6	67%	33%
43	2	21	10	40%	60%
			Average	39%	61%

It can be noticed that all students who had links recommendation after 5 initial pages visits belong to only 3 clusters. None of these students belongs to cluster 1 or cluster 4, which indicates that the frequent patterns discovered in these two clusters are short in length, or have low support values. For that reason, students belonging to these two clusters do not have any links recommended.

4.5.4. Page Recommendations after visiting 6 pages

Table 35 shows details about students who had recommendation links after visiting 6 initial pages. The numbers of links recommended to each student and the percentage of actually visited links from the recommended ones are listed in the table. A list of page IDs recommended to each student after visiting 6 pages is shown in appendix F.

Table 35: Page Recommendation after Visiting 6 Pages

Student ID	Cluster ID	Total number of pages visited	Number of recommended pages	Actually visited %	Not visited %
1	2	25	4	25%	75%
11	2	10	5	80%	20%
12	2	35	6	83%	17%
21	2	16	8	63%	38%
23	3	10	3	100%	0%
29	5	47	4	0%	100%
30	5	44	2	0%	100%
34	2	13	5	60%	40%
38	3	8	2	50%	50%
40	2	40	5	60%	40%
43	2	21	7	29%	71%
			Average	50%	50%

It can be seen from table 35 that the same observations on the experiments conducted previously about links recommendation after 3, 4, 5 initial pages visited. The number of students who had links recommendation after 6 initial pages visited decreased and the students belong to clusters 2, 3, 5.

4.5.5. Page Recommendations after visiting 7 pages

The following experiment provides details about students who had recommendation links after visiting 7 pages. Table 36 shows the same details as the previous experiments but for students who had recommendations after visiting 7 initial pages. . A list of page IDs recommended to each student after visiting 7 pages is shown in appendix G.

Table 36: Page Recommendation after Visiting 7 Pages

Student ID	Cluster ID	Total number of pages visited	Number of recommended pages	Actually visited %	Not visited %
1	2	25	2	0%	100%
11	2	10	4	75%	25%
12	2	35	4	75%	25%
21	2	16	3	67%	33%
23	3	10	1	100%	0%
34	2	13	3	33%	67%
40	2	40	4	50%	50%
43	2	21	2	50%	50%
			Average	51%	49%

It can be seen in table 36 that the students, who have links recommendation after visiting 7 pages, have relatively long navigational paths represented in the total number of pages visited column in table 36. Also, the number of recommended pages starts to decrease.

4.5.6. Page Recommendations after visiting 8 pages

Table 37 shows details about students who had recommendation links after visiting 8 initial pages. The numbers of links recommended to each student and the percentage of actually visited links from the recommended ones are listed in the table. A list of page IDs recommended to each student after visiting 8 pages is shown in appendix H.

Table 37: Page Recommendation after Visiting 8 Pages

Student ID	Cluster ID	Total number of pages visited	Number of recommended pages	Actually visited %	Not visited %
11	2	10	3	67%	33%
12	2	35	2	50%	50%
21	2	16	2	50%	50%
34	2	13	2	0%	100%
40	2	40	2	50%	50%
43	2	21	1	0%	100%
			Average	36%	64%

It can be noticed from table 37 that all students who had links recommendation after 8 page visited belong to cluster 2. Also it can be noticed, that the number of recommended pages decreased, this is normal since it is hard to discover frequent patterns of long length to be used in the recommender engine.

4.5.7. Page Recommendations after visiting 9 pages

Table 38 shows details about students who had recommendation links after visiting 9 initial pages. The numbers of links recommended to each student and the percentage of actually visited links from the recommended ones are listed in the table.

Table 38: Page Recommendation after Visiting 9 Pages

Student ID	Cluster ID	Total number of pages visited	Number of recommended pages	Actually visited %	Not visited %
11	2	10	1	0%	100%
21	2	16	1	0%	100%
			Average	0%	100%

In this experiment, only two students have recommended links, and that is logical. These two students navigated through 9 pages which are discovered to be a frequent pattern in cluster 2. Student with ID=11 in table38, has total 10 pages visited, 9 of them are recommended by the proposed technique and visited by that student.

5. Conclusions and Future Work

5.1. Conclusions

In this thesis we proposed links recommendation technique and tested it against a set of data obtained from an experiment applied at the faculty. The proposed technique proved that using clustering can help in generating recommendations to students with higher confidence since we reduced the domain. In addition, it proved that applying association rule mining on data obtained from log files can aid in generating topic related recommendations to students. The experiments conducted in this thesis, proved that the more clicks the higher the confidence of recommended links generated.

5.2.Future Work

Further research may concentrate on applying different similarity measures within clustering techniques to find more similar groups of students such as applying Jaccard measure of similarity. In addition, a comparison between the results obtained from introducing the Generalized Sequential Pattern (GSP) algorithm with the same data and the results of the Apriori algorithm applied in this thesis will be conducted.

The proposed technique will be used to find related chapters by eliminating already available structured links. In addition, further research will be conducted on the ability to restructure index pages of a website based on students' usage in different periods of time (temporal clustering).

References

Agrawal. R. , Imielinski, T. and Swami, A. (1993), Mining association rules between sets of items in large databases. , **Proceedings of the 1993ACM-SIGMODInternational conference on management of data (SIGMOD'93)**, Washington, 207–216.

Aleksandra, K. Boban, V. Mirjana, I. and Zoran, B. (2010), Integration of Recommendations and Adaptive Hypermedia into Java Tutoring System. Information Systems 2010, **OnLine-First Issue** 00, 21-21.

Brusilovsky, Peter. (1996), Methods and techniques of adaptive hypermedia. **User Modeling and User Adapted Interaction (Special issue on adaptive hypertext and hypermedia)**, 6(2-3), 87-129.

Brusillovsky, Peter (2001), Adaptive Hypermedia. **User modelling and User-Adapted Interaction**,11,87-110.

Brusilovsky, Peter (2003), Developing Adaptive Educational Hypermedia Systems: From Design Models to Authoring Tools. In: Murray, T., Blessing, S., Ainsworth, S. (eds.): **Authoring Tools for Advanced Technology Learning Environments: Toward cost-effective adaptive, interactive, and intelligent educational software**. Dordrecht, Kluwer, 377-409.

Brusilovsky, P. And Henz, N. (2007a), Open Corpus Adaptive Educational Hypermedia. In : Brusilovsky, P., Kobsa, A. and Nejdl, W. (Eds.): **The Adaptive Web Methods and Strategies of Web Personalization**, Lecture notes in computer science, Springer-Verlag, Berlin Heidelberg

Brusilovsky, P. And Millan, E. (2007b), User Models for Adaptive Hypermedia and Adaptive Educational Systems. In : Brusilovsky, P., Kobsa, A. and Nejdl, W. (Eds.): **The Adaptive Web Methods and Strategies of Web Personalization**, Springer-Verlag, Berlin Heidelberg, 3-53

Brusilovsky, P. and Peylo, C. (2003), Adaptive and Intelligent Web-based Educational Systems. **International Journal of Artificial Intelligence in Education**, 13, 156-169.

De Bra, P. And Calvia, L. (1998), AHA! An open Adaptive Hypermedia Architecture, **The New Review of Hypermedia and Multimedia**, 4, 115-139.

Gao, M. Liu, K. and Wu, Z. (2010), Personalisation in web computing and informatics: Theories, techniques, applications, and future research**. Information Systems Frontiers** , 12(5), 607-629.

Gouda, K. And Zaki, M. (2005), GenMax: An Efficient Algorithm for Mining Maximal Frequent Itemsets. **Data Mining and Knowledge Discovery**, 11,223-242.

Han, J. , Cheng, H., Xin, D and Yan, X. (2007), Frequent Pattern Mining: Current Status and Future Directions. **Data Mining and Knowledge Discovery** , 15, 55-86.

Kobsa, A., Koenemann, J. And Pohl, w. (1999) Personalized hypermedia presentation technique for improving online customer relationship. **Technical Report No**. 66, GMD, German national research center for information technology, St. Augustin, Germany

Kosala, R., and Blockeel, H. (2000), Web Mining Research: A survery**. SIGKDD Explorations**. 2(1), 1-15.

Liu, Bing. (1998), **Web Data Mining: Exploring hyperlinks, content and usage data**, (1st edition). New York: Springer

Liu, D. Lai, C. Lee, W. (2009), A hybrid of sequential rules and collaborative filtering for product recommendation. **Information Sciences**, 179,3505-3519.

Martins, A. C. Faria, L. Vaz de Carvalho, C. and Carrapatoso, E. (2008), User Modeling in Adaptive Hypermedia Educational Systems. **Educational Technology & Society**, 11 (1), 194-207.

Mobasher, Bamshad (2007), Data Mining forWeb Personalization. **The Adaptive Web**, 4321, 90-135.

Mustapaşa, O. Karahoca, A. Karahoca, D. and Uzunboylu, H. (2011), "Hello World", Web Mining for E-Learning**. Proceedings Computer Science**, 3, 1381-1387.

Raju, T. and Satyanarayana, P. S. (2008), Knowledge Discovery from Web Usage Data: Complete Preprocessing Methodology. **IJCSNS International Journal of Computer Science and Network Security**, 8(1), 179-186.

Romero, C. And Ventura, S. (2007a), Educational data mining: A survey from 1995 to 2005. **Expert Systems with Applications**, 33(1), 135-146.

Romero, C. Ventura, S. Antonio Delgado, J. and De Bra, P. (2007b), Personalized Links Recommendation Based on Data Mining in Adaptive Educational Hypermedia Systems. **Lecture Notes in Computer Science** , 4753, 292-306.

Romero, C. Ventura, S. Zafra, A. And De Bra, P. (2009), Applying Web usage mining for personalizing hyperlinks in Web-based adaptive educational systems. **Computers & Education**, 53(3), 828-840.

Sakkopoulos, E. Antoniou, D. Adamopoulou, P. Tsirakis, N. Tsakalidis, A. (2010), A web personalizing technique using adaptive data structures: The case of bursts in web visits. **Systems and Software**, 83,2200-2210.

Tan, P. Steinbach, M. And Kumar, V. (2005), **Introduction to Data Mining**, (1st edition). Boston : Addison-Wesley Longman Publishing Company.

Yongjian, F., Kanwalpreet, S., and Ming-Yi, S. (1999), “Clustering of web users based on access patterns”, In **Workshop on Web Usage Analysis and User Profiling (WEBKDD99)**, New York, USA.

Zhao, Q. and Bhowmick, S.S. Nanyang Technological University (2003), Sequential pattern mining: A survey, **(CAIS publication No. 2003118), Technical Report**, Singapore.

Appendix A: List of Pages

Page ID	Page Title
1	Reading documentation
2	Garbage collection
3	Math methods
4	Overloading
5	More printing
6	Formal and natural languages
7	What is a programming language
8	What is a program
9	What is debugging
10	Nested conditionals
11	The Node class
12	The first program
13	Converting from double to int
14	Assignment
15	Variables
16	Printing variables
17	Keywords
18	Operators
19	Copying arrays
20	Order of operations
21	Operators for Strings
22	The while statement
23	Composition
24	Local variables
25	Constructors
26	Inheritance
27	Rectangles
28	null
29	Boolean expressions
30	Two-dimensional tables
31	Methods
32	Fill-in methods
33	Sorting
34	Creating a new object
35	Incremental development vs. planning
36	Hash Functions
37	Composition
38	Abstract class
39	Shuffling
40	Veneer
41	Encapsulation and generalization
42	Shuffling and dealing
43	Subdecks

44	Decks and subdecks
45	Searching
46	The printDeck method
47	Accessing elements
48	Modifiers
49	Merge sort
50	Abstract data types
51	Type conversion
52	Invoking methods on a Graphics object
53	The return statement
54	A tree node
55	Circular buffer
56	Boolean methods
57	Invoking methods on a Graphics object
58	What's interesting
59	Length
60	Counting
61	Looping and counting
62	Run-time errors
63	Another function on Complex numbers
64	Arrays of cards
65	The fundamental ambiguity theorem
66	Statistics
67	Floating point
68	Adding new methods
69	Classes and methods
70	Programs with multiple methods
71	Methods with results
72	The modulus operator
73	Conditional execution
74	Alternative execution
75	Chained conditionals
76	Slates and Graphics objects
77	A lame Mickey Mouse
78	Other drawing commands
79	Recursion
80	A fractal Mickey Mouse
81	Stack diagrams for recursive methods
82	Convention and divine law
83	Invoking methods on objects
84	Traversal
85	The indexOf method
86	Increment and decrement operators
87	Character arithmetic
88	Pure functions
89	Encapsulation
90	Programming languages and styles
91	The LinkedList class
92	Wrappers and helpers
93	Invariants

94	The Stack ADT
95	The Java Stack Object
96	The Heap
97	Wrapper classes
98	Creating wrapper objects
99	Creating more wrapper objects
100	Getting the values out
101	Useful methods in the wrapper classes
102	The List abstract class
103	Object and class methods
104	The current object
105	Complex numbers
106	A modifier
107	Return values
108	Program development
109	Composition
110	Packages
111	Multiple assignment
112	Drawable rectangles
113	Iteration
114	Tables
115	The class hierarchy
116	Object-oriented design
117	References in objects
118	Array length
119	More encapsulation
120	More generalization
121	for loops
122	Lists as collections
123	Class definitions and object types
124	Hash table implementation
125	Resizing a hash table
126	Strings are immutable
127	Arrays and objects
128	Strings are incomparable
129	Leap of faith
130	Point objects
131	Instance variables
132	Objects as parameters
133	Objects as return types
134	Objects are mutable
135	Aliasing
136	Objects and primitives
137	Performance of resizing
138	Time
139	More constructors
140	Printing an object
141	Operations on objects
142	Which is best?
143	Generalization

144	Algorithms
145	The queue ADT
146	The Golfer class
147	Linked Queue
148	Random numbers
149	Array of random numbers
150	Many buckets
151	A single-pass solution
152	Card objects
153	The printCard method
154	The sameCard method
155	The compareCard method
156	Logical operators
157	More recursion
158	One more example
159	The toString method
160	The equals method
161	Invoking one object method from another
162	Oddities and errors
163	Lists and recursion
164	Infinite lists
165	Object methods for nodes
166	Modifying lists
167	Parsing
168	Implementing ADTs
169	Array implementation of the Stack ADT
170	Resizing arrays
171	Priority queue
172	Array implementation of Priority Queue
173	A Priority Queue client
174	Building trees
175	Traversing trees
176	Expression trees
177	Traversal
178	Defining an abstract class
179	Implementing an abstract class
180	Array implementation of trees
181	The Vector class
182	The Iterator class
183	Performance analysis
184	Analysis of mergesort
185	Overhead
186	Priority Queue implementations
187	Definition of a Heap
188	Heap remove
189	Heap insert
190	Performance of heaps
191	Heapsort
192	Arrays, Vectors and Tables
193	The Table ADT

194	The built-in Hashtable
195	A Vector implementation

Appendix B: Sample of Student/Page Visit Matrix

UID	p1	p2	p3	p4	p5	p6	p7	p8	p9	p10	p11	p12	p13	p14	p15	p16
10001	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0
10002	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10003	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10004	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	1
10005	0	0	1	1	0	0	1	1	1	1	0	1	1	0	1	0
10006	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
10007	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
10008	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
10009	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10010	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
10011	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10012	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10013	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0
10014	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
10015	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0
10016	0	0	1	0	0	0	0	1	0	1	0	0	0	0	0	0
10017	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
10018	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0
10019	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10020	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
10021	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10022	0	0	0	1	1	1	1	1	1	0	0	1	0	1	1	1
10023	0	0	0	1	0	0	1	0	0	1	0	0	0	0	0	0
10024	0	0	0	0	1	1	1	1	1	0	0	1	0	1	1	1
10025	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
10026	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0
10027	0	0	0	0	1	0	0	0	1	1	1	1	1	1	1	0
10028	0	0	0	0	1	1	1	1	1	0	0	1	0	0	0	0
10029	0	0	0	0	1	0	0	0	0	0	0	0	0	1	1	1
10030	0	0	1	0	1	0	1	0	0	1	0	0	1	1	1	1
10031	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0
10032	0	0	0	0	0	0	1	0	1	0	0	0	1	0	1	0
10033	1	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0
10034	0	0	0	1	1	0	1	0	0	0	0	0	0	1	0	0
10035	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
10036	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10037	0	0	1	0	1	1	0	1	1	0	0	1	1	1	1	1
10038	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10039	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10040	1	0	0	0	0	0	1	1	1	0	0	1	0	0	0	0
10041	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10042	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
10043	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
10044	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10045	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10046	0	0	0	1	1	0	0	0	0	0	0	0	1	0	0	0
10047	0	0	1	0	1	1	1	1	1	0	0	1	1	1	1	1
10048	0	0	0	0	1	0	0	0	0	0	1	0	0	0	1	0
10049	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

10050	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0
10051	0	0	1	1	0	0	0	0	0	1	0	0	1	0	0	0
10052	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10053	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10054	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10055	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
10056	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0
10057	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
10058	1	0	0	0	1	0	0	0	0	1	1	0	0	0	0	0
10059	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10060	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0
10061	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0
10062	0	0	0	0	0	1	0	0	0	0	0	0	1	1	1	1
10063	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10064	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10065	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
10066	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0
10067	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10068	0	0	1	1	0	0	0	0	1	0	0	0	0	0	1	0
10069	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
10070	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0
10071	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10072	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	1
10073	0	0	0	0	1	1	1	0	0	0	1	1	0	1	1	1
10074	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
10075	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
10076	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
10077	0	0	1	0	1	1	1	1	1	1	0	1	1	1	1	1
10078	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
10079	1	1	1	0	1	1	1	1	1	1	0	1	1	1	1	1
10080	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10081	1	0	1	1	1	1	1	0	0	1	0	0	1	1	1	1
10082	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0
10083	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10084	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10085	0	0	0	0	1	1	1	1	1	0	0	1	0	1	1	1
10086	0	0	0	0	1	1	0	0	0	0	0	1	0	1	1	1
10087	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
10088	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
10089	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0
10090	0	0	1	0	0	1	1	1	1	0	1	1	0	0	1	0
10091	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
10092	0	0	1	0	0	0	1	0	0	0	0	1	0	0	0	0
10093	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10094	0	0	0	0	1	1	1	1	0	0	1	1	0	1	0	1
10095	0	0	0	0	1	0	1	1	1	0	0	1	0	0	1	1
10096	0	0	1	0	0	0	1	0	0	0	0	0	0	1	0	0
10097	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1
10098	0	0	1	0	0	0	1	0	1	0	0	0	1	0	0	0
10099	0	0	1	0	1	0	0	0	0	1	0	0	1	1	0	1

Appendix C: List of Pages Recommended after 3 Pages Visited

Student ID	Student Path	Page ID recommended	Confidence
1	5,15,16	14	92.11%
1	5,15,16	17	78.95%
1	5,15,16	23	73.68%
1	5,15,16	7	71.05%
1	5,15,16	9	65.79%
1	5,15,16	6	73.68%
1	5,15,16	8	60.53%
1	5,15,16	12	73.68%
1	5,15,16	18	86.84%
1	5,15,16	20	68.42%
1	5,15,16	21	73.68%
1	5,15,16	13	52.63%
1	5,15,16	67	65.79%
1	5,15,16	3	47.37%
1	5,15,16	68	50.00%
2	121,149,151	47	87.50%
2	121,149,151	19	81.25%
2	121,149,151	127	93.75%
2	121,149,151	118	93.75%
2	121,149,151	148	81.25%
2	121,149,151	60	87.50%
2	121,149,151	66	68.75%
2	121,149,151	150	75.00%
2	121,149,151	152	68.75%
2	121,149,151	153	68.75%
3	127,121,148	47	85.71%
3	127,121,148	19	90.48%
3	127,121,148	118	95.24%
3	127,121,148	149	90.48%
3	127,121,148	60	76.19%
3	127,121,148	66	71.43%
3	127,121,148	150	57.14%
3	127,121,148	151	66.67%
3	127,121,148	64	52.38%
6	68,69,70	71	92.86%
7	3,37,68	69	100.00%
8	12,9,6	16	60.53%
8	12,9,6	7	92.11%
8	12,9,6	8	86.84%
8	12,9,6	18	63.16%
8	12,9,6	20	55.26%
8	12,9,6	21	55.26%
8	12,9,6	67	57.89%
8	12,9,6	3	47.37%
8	12,9,6	23	55.26%
8	12,9,6	5	84.21%
8	12,9,6	15	73.68%
8	12,9,6	14	71.05%
8	12,9,6	17	52.63%

11	7,8,9	16	51.28%
11	7,8,9	6	82.05%
11	7,8,9	12	84.62%
11	7,8,9	18	51.28%
11	7,8,9	20	46.15%
11	7,8,9	67	48.72%
11	7,8,9	5	74.36%
11	7,8,9	15	69.23%
11	7,8,9	14	61.54%
11	7,8,9	17	46.15%
12	7,8,9	16	51.28%
12	7,8,9	6	82.05%
12	7,8,9	12	84.62%
12	7,8,9	18	51.28%
12	7,8,9	20	46.15%
12	7,8,9	67	48.72%
12	7,8,9	5	74.36%
12	7,8,9	15	69.23%
12	7,8,9	14	61.54%
12	7,8,9	17	46.15%
13	22,114,30	120	100.00%
13	22,114,30	41	90.00%
13	22,114,30	1	90.00%
13	22,114,30	85	90.00%
13	22,114,30	87	90.00%
15	54,174,182	176	100.00%
17	7,15,67	6	91.67%
17	7,15,67	14	83.33%
17	7,15,67	23	83.33%
17	7,15,67	5	87.50%
17	7,15,67	16	79.17%
17	7,15,67	9	87.50%
17	7,15,67	8	79.17%
17	7,15,67	12	91.67%
17	7,15,67	18	83.33%
17	7,15,67	20	75.00%
17	7,15,67	13	79.17%
18	12,5,15	14	88.89%
18	12,5,15	17	66.67%
18	12,5,15	23	63.89%
18	12,5,15	16	77.78%
18	12,5,15	7	88.89%
18	12,5,15	9	83.33%
18	12,5,15	6	83.33%
18	12,5,15	8	80.56%
18	12,5,15	18	69.44%
18	12,5,15	20	66.67%
18	12,5,15	21	61.11%
18	12,5,15	13	55.56%
18	12,5,15	67	61.11%
18	12,5,15	3	50.00%
18	12,5,15	68	50.00%
19	1,85,61	83	81.82%

19	1,85,61	86	90.91%
19	1,85,61	87	81.82%
20	47,121,66	19	93.75%
20	47,121,66	127	87.50%
20	47,121,66	118	87.50%
20	47,121,66	148	87.50%
20	47,121,66	149	87.50%
20	47,121,66	60	87.50%
20	47,121,66	150	68.75%
20	47,121,66	151	75.00%
21	7,12,5	16	73.17%
21	7,12,5	9	80.49%
21	7,12,5	6	90.24%
21	7,12,5	8	78.05%
21	7,12,5	18	63.41%
21	7,12,5	20	58.54%
21	7,12,5	21	56.10%
21	7,12,5	13	46.34%
21	7,12,5	67	51.22%
21	7,12,5	68	43.90%
21	7,12,5	15	78.05%
21	7,12,5	14	80.49%
21	7,12,5	17	63.41%
21	7,12,5	23	60.98%
23	72,73,74	68	61.90%
23	72,73,74	69	80.95%
23	72,73,74	70	66.67%
23	72,73,74	71	61.90%
23	72,73,74	75	85.71%
23	72,73,74	10	85.71%
23	72,73,74	53	76.19%
23	72,73,74	51	71.43%
23	72,73,74	57	71.43%
23	72,73,74	52	66.67%
23	72,73,74	77	66.67%
23	72,73,74	78	66.67%
23	72,73,74	79	61.90%
23	72,73,74	80	66.67%
23	72,73,74	81	61.90%
23	72,73,74	107	66.67%
23	72,73,74	76	66.67%
24	140,34,132	110	90.00%
24	140,34,132	131	90.00%
24	140,34,132	134	90.00%
24	140,34,132	135	90.00%
24	140,34,132	138	90.00%
24	140,34,132	25	90.00%
24	140,34,132	88	90.00%
24	140,34,132	58	90.00%
27	47,19,121	127	88.46%
27	47,19,121	118	80.77%
27	47,19,121	148	65.38%
27	47,19,121	149	73.08%

27	47,19,121	60	57.69%
27	47,19,121	66	57.69%
27	47,19,121	150	46.15%
27	47,19,121	151	50.00%
27	47,19,121	152	46.15%
27	47,19,121	153	46.15%
27	47,19,121	64	42.31%
28	72,57,77	73	92.86%
28	72,57,77	75	100.00%
28	72,57,77	10	92.86%
28	72,57,77	53	100.00%
28	72,57,77	52	92.86%
28	72,57,77	78	92.86%
28	72,57,77	80	92.86%
28	72,57,77	81	100.00%
28	72,57,77	107	92.86%
28	72,57,77	76	92.86%
29	19,121,127	47	92.00%
29	19,121,127	118	88.00%
29	19,121,127	148	76.00%
29	19,121,127	149	84.00%
29	19,121,127	60	60.00%
29	19,121,127	66	60.00%
29	19,121,127	150	48.00%
29	19,121,127	151	56.00%
29	19,121,127	152	44.00%
29	19,121,127	64	44.00%
30	47,19,121	127	88.46%
30	47,19,121	118	80.77%
30	47,19,121	148	65.38%
30	47,19,121	149	73.08%
30	47,19,121	60	57.69%
30	47,19,121	66	57.69%
30	47,19,121	150	46.15%
30	47,19,121	151	50.00%
30	47,19,121	152	46.15%
30	47,19,121	153	46.15%
30	47,19,121	64	42.31%
31	54,174,175	176	92.86%
31	54,174,175	180	78.57%
31	54,174,175	177	92.86%
31	54,174,175	178	78.57%
32	54,174,175	176	92.86%
32	54,174,175	180	78.57%
32	54,174,175	177	92.86%
32	54,174,175	178	78.57%
33	174,176,177	54	92.31%
33	174,176,177	180	84.62%
33	174,176,177	175	100.00%
33	174,176,177	178	84.62%
34	7,5,15	14	87.88%
34	7,5,15	17	66.67%
34	7,5,15	23	66.67%

34	7,5,15	16	81.82%
34	7,5,15	9	84.85%
34	7,5,15	6	84.85%
34	7,5,15	8	81.82%
34	7,5,15	12	96.97%
34	7,5,15	18	72.73%
34	7,5,15	20	69.70%
34	7,5,15	21	66.67%
34	7,5,15	13	54.55%
34	7,5,15	67	63.64%
35	3,37,68	69	100.00%
37	3,37,69	68	82.35%
37	3,37,69	71	76.47%
37	3,37,69	74	76.47%
37	3,37,69	75	76.47%
37	3,37,69	53	76.47%
38	73,74,75	80	61.90%
38	73,74,75	37	61.90%
38	73,74,75	68	61.90%
38	73,74,75	69	80.95%
38	73,74,75	70	61.90%
38	73,74,75	51	80.95%
38	73,74,75	57	76.19%
38	73,74,75	52	71.43%
38	73,74,75	77	71.43%
38	73,74,75	78	76.19%
38	73,74,75	79	66.67%
38	73,74,75	81	61.90%
38	73,74,75	107	61.90%
38	73,74,75	4	61.90%
38	73,74,75	76	76.19%
38	73,74,75	72	85.71%
38	73,74,75	10	90.48%
38	73,74,75	53	80.95%
40	7,8,9	16	51.28%
40	7,8,9	6	82.05%
40	7,8,9	12	84.62%
40	7,8,9	18	51.28%
40	7,8,9	20	46.15%
40	7,8,9	67	48.72%
40	7,8,9	5	74.36%
40	7,8,9	15	69.23%
40	7,8,9	14	61.54%
40	7,8,9	17	46.15%
41	22,114,41	120	100.00%
41	22,114,41	119	90.00%
41	22,114,41	30	90.00%
41	22,114,41	85	100.00%
41	22,114,41	87	90.00%
43	15,14,16	17	82.05%
43	15,14,16	23	79.49%
43	15,14,16	5	89.74%
43	15,14,16	7	64.10%

43	15,14,16	9	58.97%
43	15,14,16	6	74.36%
43	15,14,16	8	53.85%
43	15,14,16	12	66.67%
43	15,14,16	18	84.62%
43	15,14,16	20	66.67%
43	15,14,16	21	71.79%
43	15,14,16	13	56.41%
43	15,14,16	67	66.67%
43	15,14,16	3	46.15%
43	15,14,16	68	48.72%

Appendix D: List of Pages Recommended after 4 Pages Visited

Student ID	Student Path	Page ID recommended	Confidence
1	5,15,16,17	14	100.00%
1	5,15,16,17	23	80.00%
1	5,15,16,17	7	70.00%
1	5,15,16,17	9	66.67%
1	5,15,16,17	6	80.00%
1	5,15,16,17	8	63.33%
1	5,15,16,17	12	73.33%
1	5,15,16,17	18	93.33%
1	5,15,16,17	20	76.67%
1	5,15,16,17	21	76.67%
1	5,15,16,17	67	66.67%
2	121,149,151,150	47	91.67%
2	121,149,151,150	19	91.67%
2	121,149,151,150	127	100.00%
2	121,149,151,150	118	91.67%
2	121,149,151,150	148	91.67%
2	121,149,151,150	60	100.00%
3	127,121,148,66	47	93.33%
3	127,121,148,66	19	100.00%
3	127,121,148,66	118	93.33%
3	127,121,148,66	149	93.33%
3	127,121,148,66	60	80.00%
3	127,121,148,66	151	73.33%
8	12,9,6,17	15	90.00%
8	12,9,6,17	14	100.00%
8	12,9,6,17	5	100.00%
8	12,9,6,17	16	100.00%
8	12,9,6,17	7	95.00%
8	12,9,6,17	8	90.00%
8	12,9,6,17	18	95.00%
8	12,9,6,17	20	90.00%
11	7,8,9,6	5	84.38%
11	7,8,9,6	16	59.38%
11	7,8,9,6	12	96.88%
11	7,8,9,6	18	62.50%
11	7,8,9,6	20	56.25%
11	7,8,9,6	67	59.38%
11	7,8,9,6	15	78.13%
11	7,8,9,6	14	71.88%
12	7,8,9,6	5	84.38%
12	7,8,9,6	16	59.38%
12	7,8,9,6	12	96.88%
12	7,8,9,6	18	62.50%
12	7,8,9,6	20	56.25%
12	7,8,9,6	67	59.38%
12	7,8,9,6	15	78.13%
12	7,8,9,6	14	71.88%
13	22,114,30,41	87	100.00%
13	22,114,30,41	85	100.00%

13	22,114,30,41	120	100.00%
18	12,5,15,14	17	75.00%
18	12,5,15,14	23	71.88%
18	12,5,15,14	16	78.13%
18	12,5,15,14	7	87.50%
18	12,5,15,14	9	84.38%
18	12,5,15,14	6	87.50%
18	12,5,15,14	8	81.25%
18	12,5,15,14	18	75.00%
18	12,5,15,14	20	75.00%
18	12,5,15,14	21	65.63%
18	12,5,15,14	13	62.50%
18	12,5,15,14	67	65.63%
18	12,5,15,14	3	56.25%
19	1,85,61,86	87	90.00%
20	47,121,66,60	19	92.86%
20	47,121,66,60	127	85.71%
20	47,121,66,60	118	85.71%
20	47,121,66,60	148	85.71%
20	47,121,66,60	149	85.71%
20	47,121,66,60	150	78.57%
20	47,121,66,60	151	78.57%
21	7,12,5,15	14	87.50%
21	7,12,5,15	17	65.63%
21	7,12,5,15	23	65.63%
21	7,12,5,15	16	81.25%
21	7,12,5,15	9	87.50%
21	7,12,5,15	6	87.50%
21	7,12,5,15	8	84.38%
21	7,12,5,15	18	71.88%
21	7,12,5,15	20	68.75%
21	7,12,5,15	21	65.63%
21	7,12,5,15	67	62.50%
23	72,73,74,75	69	83.33%
23	72,73,74,75	70	72.22%
23	72,73,74,75	10	88.89%
23	72,73,74,75	53	88.89%
23	72,73,74,75	51	77.78%
23	72,73,74,75	57	83.33%
23	72,73,74,75	52	72.22%
23	72,73,74,75	78	72.22%
23	72,73,74,75	81	72.22%
23	72,73,74,75	107	72.22%
23	72,73,74,75	76	72.22%
27	47,19,121,118	127	95.24%
27	47,19,121,118	148	76.19%
27	47,19,121,118	149	85.71%
27	47,19,121,118	60	61.90%
27	47,19,121,118	66	61.90%
27	47,19,121,118	151	57.14%
28	72,57,77,78	75	100.00%
28	72,57,77,78	10	100.00%
28	72,57,77,78	53	100.00%

28	72,57,77,78	52	100.00%
28	72,57,77,78	81	100.00%
29	19,121,127,148	47	89.47%
29	19,121,127,148	118	94.74%
29	19,121,127,148	149	94.74%
29	19,121,127,148	60	78.95%
29	19,121,127,148	66	78.95%
29	19,121,127,148	150	63.16%
29	19,121,127,148	151	68.42%
30	47,19,121,127	118	86.96%
30	47,19,121,127	148	73.91%
30	47,19,121,127	149	82.61%
30	47,19,121,127	60	60.87%
30	47,19,121,127	66	60.87%
30	47,19,121,127	150	47.83%
30	47,19,121,127	151	56.52%
30	47,19,121,127	152	47.83%
32	54,174,175,176	177	92.31%
34	7,5,15,14	17	75.86%
34	7,5,15,14	23	75.86%
34	7,5,15,14	16	82.76%
34	7,5,15,14	9	86.21%
34	7,5,15,14	6	89.66%
34	7,5,15,14	8	82.76%
34	7,5,15,14	12	96.55%
34	7,5,15,14	18	79.31%
34	7,5,15,14	20	79.31%
34	7,5,15,14	21	72.41%
34	7,5,15,14	13	62.07%
34	7,5,15,14	67	68.97%
38	73,74,75,10	68	68.42%
38	73,74,75,10	69	78.95%
38	73,74,75,10	72	84.21%
38	73,74,75,10	53	84.21%
38	73,74,75,10	51	89.47%
38	73,74,75,10	57	78.95%
38	73,74,75,10	52	78.95%
38	73,74,75,10	77	68.42%
38	73,74,75,10	78	78.95%
38	73,74,75,10	79	68.42%
38	73,74,75,10	76	78.95%
40	7,8,9,6	5	84.38%
40	7,8,9,6	16	59.38%
40	7,8,9,6	12	96.88%
40	7,8,9,6	18	62.50%
40	7,8,9,6	20	56.25%
40	7,8,9,6	67	59.38%
40	7,8,9,6	15	78.13%
40	7,8,9,6	14	71.88%
43	15,14,16,17	23	81.25%
43	15,14,16,17	5	93.75%
43	15,14,16,17	7	65.63%
43	15,14,16,17	9	62.50%

43	15,14,16,17	6	78.13%
43	15,14,16,17	8	59.38%
43	15,14,16,17	12	71.88%
43	15,14,16,17	18	90.63%
43	15,14,16,17	20	71.88%
43	15,14,16,17	21	75.00%
43	15,14,16,17	13	59.38%
43	15,14,16,17	67	68.75%

Appendix E: List of Pages Recommended after 5 Pages Visited

Student ID	Student Path	Page ID recommended	Confidence
1	5,15,16,17,18	23	82.14%
1	5,15,16,17,18	7	71.43%
1	5,15,16,17,18	9	67.86%
1	5,15,16,17,18	6	78.57%
1	5,15,16,17,18	8	64.29%
1	5,15,16,17,18	12	71.43%
1	5,15,16,17,18	20	78.57%
1	5,15,16,17,18	21	78.57%
1	5,15,16,17,18	67	67.86%
1	5,15,16,17,18	14	100.00%
3	127,121,148,66,60	47	100.00%
3	127,121,148,66,60	19	100.00%
3	127,121,148,66,60	118	91.67%
3	127,121,148,66,60	149	91.67%
8	12,9,6,17,18	14	100.00%
8	12,9,6,17,18	5	100.00%
8	12,9,6,17,18	16	100.00%
8	12,9,6,17,18	7	94.74%
11	7,8,9,6,12	15	80.65%
11	7,8,9,6,12	14	74.19%
11	7,8,9,6,12	5	87.10%
11	7,8,9,6,12	16	61.29%
11	7,8,9,6,12	18	64.52%
11	7,8,9,6,12	20	58.06%
11	7,8,9,6,12	67	61.29%
12	7,8,9,6,12	15	80.65%
12	7,8,9,6,12	14	74.19%
12	7,8,9,6,12	5	87.10%
12	7,8,9,6,12	16	61.29%
12	7,8,9,6,12	18	64.52%
12	7,8,9,6,12	20	58.06%
12	7,8,9,6,12	67	61.29%
18	12,5,15,14,16	7	92.00%
18	12,5,15,14,16	9	84.00%
18	12,5,15,14,16	6	96.00%
18	12,5,15,14,16	8	80.00%
18	12,5,15,14,16	18	88.00%
18	12,5,15,14,16	20	92.00%
18	12,5,15,14,16	21	84.00%
18	12,5,15,14,16	67	72.00%
18	12,5,15,14,16	17	88.00%
18	12,5,15,14,16	23	80.00%
21	7,12,5,15,14	16	82.14%
21	7,12,5,15,14	9	89.29%
21	7,12,5,15,14	6	92.86%
21	7,12,5,15,14	8	85.71%
21	7,12,5,15,14	18	78.57%
21	7,12,5,15,14	20	78.57%
21	7,12,5,15,14	21	71.43%

21	7,12,5,15,14	67	67.86%
21	7,12,5,15,14	17	75.00%
21	7,12,5,15,14	23	75.00%
23	72,73,74,75,10	69	81.25%
23	72,73,74,75,10	53	93.75%
23	72,73,74,75,10	51	87.50%
23	72,73,74,75,10	57	87.50%
23	72,73,74,75,10	52	81.25%
27	47,19,121,118,60	127	100.00%
27	47,19,121,118,60	148	100.00%
27	47,19,121,118,60	149	100.00%
27	47,19,121,118,60	66	84.62%
29	19,121,127,148,60	47	93.33%
29	19,121,127,148,60	118	93.33%
29	19,121,127,148,60	149	93.33%
29	19,121,127,148,60	66	80.00%
29	19,121,127,148,60	150	80.00%
29	19,121,127,148,60	151	80.00%
30	47,19,121,127,149	118	94.74%
30	47,19,121,127,149	148	84.21%
30	47,19,121,127,149	60	68.42%
30	47,19,121,127,149	66	68.42%
30	47,19,121,127,149	151	63.16%
34	7,5,15,14,16	9	83.33%
34	7,5,15,14,16	6	91.67%
34	7,5,15,14,16	8	79.17%
34	7,5,15,14,16	12	95.83%
34	7,5,15,14,16	18	91.67%
34	7,5,15,14,16	20	95.83%
34	7,5,15,14,16	21	87.50%
34	7,5,15,14,16	67	75.00%
34	7,5,15,14,16	17	87.50%
34	7,5,15,14,16	23	83.33%
38	73,74,75,10,51	69	76.47%
38	73,74,75,10,51	72	82.35%
38	73,74,75,10,51	53	82.35%
38	73,74,75,10,51	57	82.35%
38	73,74,75,10,51	52	88.24%
38	73,74,75,10,51	77	76.47%
38	73,74,75,10,51	78	82.35%
38	73,74,75,10,51	76	88.24%
40	7,8,9,6,5	15	85.19%
40	7,8,9,6,5	14	85.19%
40	7,8,9,6,5	16	70.37%
40	7,8,9,6,5	12	100.00%
40	7,8,9,6,5	18	70.37%
40	7,8,9,6,5	20	66.67%
43	15,14,16,17,18	23	82.76%
43	15,14,16,17,18	5	96.55%
43	15,14,16,17,18	7	68.97%
43	15,14,16,17,18	9	65.52%
43	15,14,16,17,18	6	79.31%
43	15,14,16,17,18	8	62.07%

43	15,14,16,17,18	12	68.97%
43	15,14,16,17,18	20	75.86%
43	15,14,16,17,18	21	79.31%
43	15,14,16,17,18	67	68.97%

Appendix F: List of Pages Recommended after 6 Pages Visited

Student ID	Student Path	Page ID recommended	Confidence
1	5,15,16,17,18,21	14	100.00%
1	5,15,16,17,18,21	23	90.91%
1	5,15,16,17,18,21	6	86.36%
1	5,15,16,17,18,21	20	90.91%
11	7,8,9,6,12,5	15	85.19%
11	7,8,9,6,12,5	14	85.19%
11	7,8,9,6,12,5	16	70.37%
11	7,8,9,6,12,5	18	70.37%
11	7,8,9,6,12,5	20	66.67%
12	7,8,9,6,12,15	14	88.00%
12	7,8,9,6,12,15	5	92.00%
12	7,8,9,6,12,15	16	72.00%
12	7,8,9,6,12,15	18	76.00%
12	7,8,9,6,12,15	20	72.00%
12	7,8,9,6,12,15	67	72.00%
21	7,12,5,15,14,16	17	86.96%
21	7,12,5,15,14,16	23	82.61%
21	7,12,5,15,14,16	9	86.96%
21	7,12,5,15,14,16	6	95.65%
21	7,12,5,15,14,16	8	82.61%
21	7,12,5,15,14,16	18	91.30%
21	7,12,5,15,14,16	20	95.65%
21	7,12,5,15,14,16	21	86.96%
23	72,73,74,75,10,53	51	86.67%
23	72,73,74,75,10,53	57	86.67%
23	72,73,74,75,10,53	52	86.67%
29	19,121,127,148,60,151	47	91.67%
29	19,121,127,148,60,151	118	91.67%
29	19,121,127,148,60,151	149	91.67%
29	19,121,127,148,60,151	150	100.00%
30	47,19,121,127,149,151	118	100.00%
30	47,19,121,127,149,151	148	91.67%
34	7,5,15,14,16,17	6	90.48%
34	7,5,15,14,16,17	12	95.24%
34	7,5,15,14,16,17	18	95.24%
34	7,5,15,14,16,17	20	95.24%
34	7,5,15,14,16,17	21	85.71%
38	73,74,75,10,51,76	57	86.67%
38	73,74,75,10,51,76	52	86.67%
40	7,8,9,6,5,15	14	95.65%
40	7,8,9,6,5,15	16	78.26%
40	7,8,9,6,5,15	12	100.00%
40	7,8,9,6,5,15	18	78.26%
40	7,8,9,6,5,15	20	78.26%
43	15,14,16,17,18,20	23	86.36%
43	15,14,16,17,18,20	5	100.00%
43	15,14,16,17,18,20	7	86.36%
43	15,14,16,17,18,20	9	81.82%
43	15,14,16,17,18,20	6	86.36%

43	15,14,16,17,18,20	12	86.36%
43	15,14,16,17,18,20	21	90.91%

Appendix G: List of Pages Recommended after 7 Pages Visited

Student ID	Student Path	Page ID recommended	Confidence
1	5,15,16,17,18,21,23	14	100.00%
1	5,15,16,17,18,21,23	20	90.00%
11	7,8,9,6,12,5,15	14	95.65%
11	7,8,9,6,12,5,15	16	78.26%
11	7,8,9,6,12,5,15	18	78.26%
11	7,8,9,6,12,5,15	20	78.26%
12	7,8,9,6,12,15,14	5	100.00%
12	7,8,9,6,12,15,14	16	81.82%
12	7,8,9,6,12,15,14	18	81.82%
12	7,8,9,6,12,15,14	20	81.82%
21	7,12,5,15,14,16,17	6	95.00%
21	7,12,5,15,14,16,17	18	95.00%
21	7,12,5,15,14,16,17	20	95.00%
23	72,73,74,75,10,53,51	52	100.00%
34	7,5,15,14,16,17,18	6	90.00%
34	7,5,15,14,16,17,18	12	95.00%
34	7,5,15,14,16,17,18	20	95.00%
40	7,8,9,6,5,15,14	16	81.82%
40	7,8,9,6,5,15,14	12	100.00%
40	7,8,9,6,5,15,14	18	81.82%
40	7,8,9,6,5,15,14	20	81.82%
43	15,14,16,17,18,20,21	23	90.00%
43	15,14,16,17,18,20,21	5	100.00%

Appendix H: List of Pages Recommended after 8 Pages Visited

Student ID	Student Path	Page ID recommended	Confidence
11	7,8,9,6,12,5,15,14	16	81.82%
11	7,8,9,6,12,5,15,14	18	81.82%
11	7,8,9,6,12,5,15,14	20	81.82%
12	7,8,9,6,12,15,14,16	5	100.00%
12	7,8,9,6,12,15,14,16	20	100.00%
21	7,12,5,15,14,16,17,18	6	94.74%
21	7,12,5,15,14,16,17,18	20	94.74%
34	7,5,15,14,16,17,18,20	6	94.74%
34	7,5,15,14,16,17,18,20	12	94.74%
40	7,8,9,6,5,15,14,16	12	100.00%
40	7,8,9,6,5,15,14,16	20	100.00%
43	15,14,16,17,18,20,21,23	5	100.00%
11	7,8,9,6,12,5,15,14	16	81.82%

إكتشاف نمطية الاستخدام واقتراح الروابط الملائمة في الهابيرميديا التعليمية المتأقلمة

اعداد
ريم علي محمد القعدان الفايز

المشرف
الدكتور عمار الحنيطي

ملخص

أصبح الانترنت واحد من اكثر الوسائل المنتشرة و المفضلة للتعليم. معظم النظم التعليمية الحالية الموجودة على شبكة الإنترنت لا تزال تطرح نفس الروابط لجميع الطلاب بغض النظر عن زياراتهم السابقة لهذه الانظمة. عدم وجود توجيهات مخصصة للطلاب أثناء تصفحهم انظمة الويب سوف تحد من قدرتهم على المشاركة والاستفادة من المواد المنشورة في الوسائط صفحات الهايبرميديا.

الهدف من هذه الأطروحة هو الاستفادة من استخدام البيانات الناتجة عن تفاعل الطلاب السابق مع الوسائط المتعددة التعليمية عبر الإنترنت من أجل اكتشاف أنماط مثيرة للاهتمام في هذه البيانات. تستخدم هذه الأنماط لتخصيص وتنظيم صفحات الهايبرميديا وفقا لتصفح كل طالب السابق.

هذا البحث يطرح مزيج من طرق التنقيب عن البيانات من اجل تقديم اقتراحات مرتبطة المواضيع للطلاب استنادا على التصفح السابق للطلاب الآخرين. تم اختبار هذه الطريقة على البيانات الناتجة عن تفاعل مجموعة من الطلاب مع نظام تعليمي على الانترنت لتعليم الجافا وكشفت التجارب أن الطريقة المقترحه لديها القدرة على اقتراح روابط مرتبطة المواضيع للطلاب الذين يتصفحون الهايبرميديا.